



UNIVERSIDADE DO VALE DO TAQUARI – UNIVATES  
CURSO DE SISTEMAS DE INFORMAÇÃO

**A INFLUÊNCIA DA COLABORAÇÃO INTEGRADA AO AMBIENTE  
DE DESENVOLVIMENTO NA APRENDIZAGEM DA LÓGICA DE  
PROGRAMAÇÃO**

Tainá Schaeffer

Lajeado, novembro de 2020

Tainá Schaeffer

**A INFLUÊNCIA DA COLABORAÇÃO INTEGRADA AO  
AMBIENTE DE DESENVOLVIMENTO NA  
APRENDIZAGEM DA LÓGICA DE PROGRAMAÇÃO**

Monografia apresentada ao Centro de Ciências  
Exatas e Tecnológicas da Universidade do  
Vale do Taquari - UNIVATES, como parte dos  
requisitos para obtenção de título de Bacharel  
em Sistemas de Informação.

Orientador: Prof. Me. Juliano Dertzbacher

Lajeado, novembro de 2020

## RESUMO

Os recursos tecnológicos estão sendo desenvolvidos e aprimorando a forma como nos comunicamos ou nos mantemos informados, gerando impactos no dia-a-dia. Esta crescente demanda por novas e avançadas ferramentas exige a formação de profissionais qualificados e que estejam constantemente atualizando seus conhecimentos para desenvolverem ferramentas que atendam essas necessidades. O aprendizado da lógica de programação é o primeiro passo para formar profissionais de Tecnologia da Informação, porém, faltam recursos que facilitem efetivamente a aprendizagem dos estudantes. Uma forma de auxiliá-los é promover o aprendizado em grupos de colaboração, onde os estudantes com maior facilidade de aprendizado podem auxiliar aqueles que possuem alguma dificuldade. Diante disso, este trabalho busca verificar se a utilização de uma ferramenta integrada ao ambiente de desenvolvimento auxilia no aprendizado de lógica de programação e, desta forma, uma ferramenta de comunicação foi desenvolvida e disponibilizada aos estudantes, oferecendo ao professor mecanismos de mineração de textos, permitindo analisar o comportamento e a colaboração de seus alunos em sala de aula. Como resultado, foi possível comprovar que os recursos implementados facilitam o aprendizado e é possível expandir o uso da aplicação para outras finalidades.

**Palavras-chave:** Lógica de programação. Aprendizagem colaborativa. Grupos de colaboração. Ambiente de desenvolvimento colaborativo. Mineração de textos.

## **ABSTRACT**

The technological resources are being developed and improving the way how we communicate or keep informed, generating impacts day by day. This increasing demand for new and advanced tools requires the formation of skilled professionals and who are constantly updating their knowledge to develop tools that meet these needs. The learning of programming logic is the first step to train professionals of Information Technology, but are scarce resources that effectively facilitate the learning of students. One way to help them is promoting the learning in groups of collaboration, where students with greater ease of learning can help those who have some difficulty. At that, this work seeks to verify if the use of a tool integrated development environment helps in the learning of programming logic and, in this, way, a communication tool was developed and made available to the students, offering the teacher text mining mechanisms, allowing the analysis of the behavior and collaboration of their students in the classroom. As a result, it was possible to prove that the implemented resources facilitate learning and it is possible to expand the use of application for other purposes.

**Keywords:** Programming logic. Collaborative learning. Groups to collaborate. Collaborative development environment. Text mining.

## LISTA DE FIGURAS

Figura 1 - Diagrama do processo de programação .....	15
Figura 2 - Diagrama das áreas de descoberta do conhecimento .....	20
Figura 3 – Processo de descoberta de conhecimento em textos. ....	23
Figura 4 - Arquitetura da ferramenta .....	25
Figura 5 - Estrutura computacional do APP .....	26
Figura 6 - Interface do MineraFórum .....	28
Figura 7 - Fluxograma do desenvolvimento do trabalho. ....	33
Figura 8 - Tabela de probabilidade Naive Bayes.....	38
Figura 9 - Probabilidade do Naive Bayes. ....	39
Figura 10 - Classificador de Naive Bayes.....	39
Figura 11 - Modelo de caso de uso da plataforma para professores e administradores.....	43
Figura 12 - Modelo de caso de uso da plataforma para alunos.....	43
Figura 13 - Modelo do banco de dados.....	44
Figura 14 - Tela de listagem de discussões do grupo .....	46
Figura 15 – Tela de discussões .....	47
Figura 16 – Visualização da listagem de discussões do EaseChat web.....	47
Figura 17 - Processamento dos dados do grupo no minerador de textos.....	48
Figura 18 - Busca dos dados do grupo no Dashboard .....	49
Figura 19 - Indicadores do Dashboard.....	49
Figura 20 - Relatório de análise de solicitude.....	50
Figura 21 - Listagem de perguntas.....	51
Figura 22 - Processo de remoção de <i>stopwords</i> .....	55
Figura 23 – Aplicação da técnica <i>Stemming</i> .....	55
Figura 24 – Total de mensagens classificadas como solícitas e não solícitas. ....	56

Figura 25 – Acurácia do algoritmo. ....	56
Figura 26 – Análise de solicitude do grupo Águia Azul.....	57
Figura 27 - Análise de solicitude do grupo Falcão Negro. ....	58
Figura 28 - Análise de solicitude do grupo “Mamute Dourado”. ....	58
Figura 29 - Análise de solicitude do grupo Onça Pintada. ....	59
Figura 30 - Análise de solicitude do grupo PI1 – 1. ....	60
Figura 31 - Análise de solicitude do grupo PI1 – 2. ....	60
Figura 32 - Análise de solicitude do grupo PI1 – 3. ....	61
Figura 33 - Análise de solicitude do grupo PI1 – 4. ....	62
Figura 34 - Questão um do questionário, referente a importância de ferramentas colaborativas. ....	63
Figura 35 - Questão dois, referente a eficácia da ferramenta.....	63
Figura 36 - Questão três, referente ao uso a longo prazo da ferramenta colaborativa.....	64
Figura 37 - Questão quatro, referente ao aprendizado em conjunto. ....	64
Figura 38 - Questão cinco, referente aos pontos positivos da ferramenta. ....	65
Figura 39 - Questão seis, referente aos pontos negativos da ferramenta. ....	66

## **LISTA DE QUADROS**

Quadro 1 - Elementos que compõem a aprendizagem colaborativa.....	17
Quadro 2 – Detalhamento do processo de descoberta de conhecimento em textos.....	23
Quadro 3 - Lista de requisitos funcionais .....	40
Quadro 4 - Lista de requisitos não funcionais .....	41

## **LISTA DE ABREVIATURAS**

TI - Tecnologia da Informação

TIC - Tecnologia da Informação e Comunicação

IDE - Integrated Development Environment

KDD - Knowledge Discovery in Databases

KDT - Knowledge Discovery from Text

POO – Programação Orientada à Objetos

HTML - Hiper Text Markup Language

CSS – Cascading Style Sheets

SGBD - Sistema Gerenciador de Banco de dados

WS - Web Service

PLN - Processamento de Linguagem Natural

NLTK - Natural Language Toolkit



## SUMÁRIO

<b>1 INTRODUÇÃO.....</b>	<b>11</b>
1.1 Problema de pesquisa .....	12
1.2 Objetivos .....	13
1.3 Estrutura do trabalho.....	13
<b>2 FUNDAMENTAÇÃO TEÓRICA .....</b>	<b>14</b>
2.1 Dimensão pedagógica.....	14
2.1.1 Aprendizagem de lógica de programação.....	14
2.1.2 Aprendizagem colaborativa .....	16
2.1.3 Trabalhos em grupo de colaboração .....	17
2.2 Dimensão tecnológica.....	19
2.2.1 Sistemas de informação .....	19
2.2.2 Ambiente colaborativo.....	19
2.2.3 Mineração de textos .....	20
2.2.4 Algoritmos de mineração de textos. ....	22
2.2.5 Etapas da mineração de textos.....	23
2.2.6 Processamento de linguagem natural.....	24
<b>3 TRABALHOS RELACIONADOS.....</b>	<b>25</b>
3.1 Ferramenta colaborativa para apoiar a aprendizagem de programação de computadores .....	25
3.2 APP- Uma arquitetura pedagógica para aprendizagem de programação de computadores .....	27
3.3 Comunicação, interação e aprendizagem: o fórum de discussão como estratégia de ensino.....	28
3.4 Análise das mensagens de fóruns de discussão através de um software para mineração de textos.....	29
3.5 Aplicação da mineração de textos em fóruns de discussão .....	30
<b>4 MATERIAIS E MÉTODOS .....</b>	<b>32</b>
4.1 Tipo de pesquisa.....	32
4.2 Tecnologias utilizadas.....	34
4.2.1 Apache.....	34
4.2.2 HTML e CSS .....	34
4.2.3 PHP.....	35
4.2.4 Adianti Framework .....	35

4.2.5 PostgreSQL.....	36
4.2.6 NetBeans IDE .....	36
4.2.7 Java.....	36
4.2.8 Web Services.....	37
4.2.9 Python. ....	37
4.2.10 Bibliotecas Python.....	38
4.2.11 Naive Bayes.....	38
4.3 Desenvolvimento do projeto .....	40
4.3.1 Requisitos.....	40
4.3.1.1 Requisitos funcionais .....	40
4.3.1.2 Requisitos não funcionais .....	41
4.3.2 Modelo de casos de uso .....	42
4.3.3 Modelo relacional.....	44
4.3.4 Interface da ferramenta .....	46
 5 RESULTADOS E DISCUSSÕES.....	 53
5.1 Coleta dos dados.....	54
5.2 Pré-processamento e classificação. ....	54
5.3 Análise dos dados. ....	56
5.4 Análise dos dados da turma de Programação Orientada a Objetos. ....	57
5.5 Análise dos dados da turma de Projeto Integrador I. ....	60
5.6 Análise do questionário aplicado.....	63
 6 CONCLUSÕES.....	 68
 REFERÊNCIAS.....	 70
 APÊNDICE A – GUIA PARA INSTALAÇÃO E UTILIZAÇÃO DA FERRAMENTA “EASECHAT” .....	  74
 APÊNDICE B – QUESTIONÁRIO DE AVALIAÇÃO DO “EASECHAT”. ..	 82

# **1 INTRODUÇÃO**

Vivemos em uma constante evolução, principalmente no que diz respeito à tecnologia. A forma com que nos comunicamos ou nos mantemos informados passou a ser instantânea, gerando grandes impactos no nosso dia-a-dia.

Lara (2019) reconhece que os sistemas estão em toda parte e que este termo está fortemente relacionado à área de tecnologia. Para Chiavenato (2010) a definição de sistema está diretamente relacionada com a ideia de totalidade e integração. Suas partes são compostas por uma rede de interações com a finalidade de atingir um objetivo específico. Segundo o autor, as tecnologias deixaram de ser vistas como ferramentas isoladas e passaram a ser tratadas como ferramentas formadas por elementos dinâmicos que, por meio de suas interações, buscam alcançar objetivos.

Neste contexto, aumenta a demanda por profissionais de Tecnologia da Informação (TI) capacitados a trabalhar nas novas demandas do mercado. O aprendizado da lógica de programação é um dos primeiros passos para a formação de profissionais de TI e, para muitos, é um dilema e o fator mais significativo para a evasão de cursos da área de TI. Puga e Riseti (2017) afirmam que o aprendizado da lógica possibilita a organização do raciocínio, a compreensão de conceitos e o melhor entendimento de conteúdos mais avançados. Para Noschang, Pelz e Raabe (2014) os conhecimentos consolidados pelos estudantes nos semestres iniciais da formação influencia diretamente no desempenho dos mesmos durante a trajetória acadêmica.

Ao longo do tempo algumas técnicas e ferramentas foram desenvolvidas com o objetivo de facilitar a aprendizagem da lógica de programação tendo como finalidade obter uma melhora no desempenho da aprendizagem dos estudantes. Entretanto, vemos que ainda podem ser

desenvolvidos novos recursos para facilitar o aprendizado.

Para Ribeiro (2012), estudar em grupos é a melhor forma de compartilhar conhecimentos e habilidades de forma mais rápida e profunda. Segundo o autor, trabalhar em grupos é importante, pois provoca questionamentos, a realização de debates e a formação do pensamento crítico. De modo geral, as organizações influenciam os seus profissionais a desenvolverem uma postura colaborativa a fim de que auxiliem para o desenvolvimento de competências pessoais e de grupo.

Na próxima seção é apresentado o problema de pesquisa que guiou o desenvolvimento deste trabalho.

## **1.1 Problema de pesquisa**

O objetivo de uma *Integrated Development Environment* (IDE) é reunir as atividades realizadas pelos desenvolvedores de software durante a implementação. De modo geral, a utilização de uma IDE possibilita uma maior produtividade do desenvolvedor. Entretanto, a maioria delas não possuem mecanismos que proporcionem o compartilhamento de problemas e dúvidas ou que favorecem a contribuição para o aprendizado de outros usuários.

Esse tipo de carência faz com que os usuários tenham que buscar maneiras alternativas de sanarem as suas dúvidas em fóruns de discussão na Internet. Isto pode fazer com que a pesquisa demore mais tempo para ser feita e, em muitos casos, pode ser que o usuário não obtenha a resposta que necessita.

Diante disso, o presente trabalho se propôs a responder a seguinte questão de pesquisa: a possibilidade de trocar mensagens com os colegas de grupo, de forma integrada ao ambiente de desenvolvimento, auxilia a colaboração para aprendizagem de programação?

## **1.2 Objetivos**

Considerando que um dos elementos com maior relevância para a aprendizagem colaborativa é a comunicação, o presente trabalho teve como objetivo geral verificar se a colaboração entre grupos, integrada ao ambiente de desenvolvimento, favorece o aprendizado de lógica de programação. Para isso foi desenvolvida uma solução tecnológica capaz de realizar a comunicação dos estudantes diretamente no ambiente de desenvolvimento de Software, oferecendo recursos para o compartilhamento de dúvidas e soluções para problemas. Desta forma, é possível verificar as interações feitas entre os estudantes por meio da mineração de textos e identificar a postura dos estudantes para determinar se alunos colaborativos cooperam na construção dos conhecimentos de outros estudantes.

Para que o objetivo geral possa ser alcançado, os objetivos específicos identificados na sequência, também precisam ser considerados:

- a) Analisar a relevância de uma ferramenta para comunicação, integrada à uma IDE;
- b) Avaliar e aplicar algoritmos de reconhecimento de padrões de escrita de textos para identificar os usuários com perfil colaborativo;
- c) Fornecer aos professores mecanismos que possibilitem monitorar contribuir para a interação entre os estudantes.

## **1.3 Estrutura do trabalho**

O presente trabalho está dividido em seis seções. A primeira seção apresenta a visão geral do trabalho, o problema de pesquisa, os objetivos e a estrutura do trabalho. Na segunda seção são detalhados os conceitos relacionados à elaboração deste trabalho. A terceira seção apresenta trabalhos relacionados. Na quarta seção são descritas as metodologias e ferramentas utilizadas para o desenvolvimento do trabalho. Na seção cinco são realizadas as considerações finais. Por fim, são apresentadas as referências utilizadas.

## **2 FUNDAMENTAÇÃO TEÓRICA**

O presente capítulo tem como objetivo contextualizar os assuntos relacionados à compreensão teórica dos fundamentos do problema de pesquisa. Desta forma, este capítulo está separado em duas partes: dimensão pedagógica, onde serão abordados assuntos como aprendizagem de lógica de programação, postura interpessoal e aprendizado em grupos e; dimensão tecnológica, que abordará assuntos como sistemas de informação, ambientes colaborativos e mineração de textos.

### **2.1 Dimensão pedagógica**

A dimensão pedagógica possui um viés que considera atividades com potencial para serem aplicadas pelo professor objetivando possibilitar a discussão entre seus alunos, facilitando a pesquisa e a discussão de temáticas em diferentes pontos de vista. A seguir serão abordados assuntos que colaboram para o desenvolvimento dos estudantes sobre a perspectiva da aprendizagem de lógica de programação através de grupos de colaboração.

#### **2.1.1 Aprendizagem de lógica de programação**

Puga e Riseti (2017) definem a lógica, associada à programação, como a organização e planejamento de instruções assertivas em um algoritmo com a finalidade de implementar sua viabilidade em um programa. Segundo os autores, o estudo da lógica pode ser definido em

indutivo e dedutivo onde ambos se baseiam em um conjunto de enunciados formado por premissas e conclusão.

Segundo Guedes (2014) a programação é feita através de um processo onde devem ser considerados vários elementos. O autor define a lógica de programação como um meio para buscar uma solução computacional conforme é possível analisar na Figura 1.

Figura 1 - Diagrama do processo de programação



Fonte: Guedes (2014, p. 117)

Para Forbellone e Eberspacher (2013) lógica de programação significa o uso adequado da linha de pensamento, da razão e do raciocínio através da representação formal da programação, com a finalidade de atingir à racionalidade e o desenvolvimento de técnicas que assessorem na solução de problemas. Para eles, a lógica de programação pode ser concebida pela mente treinada e pode ser representada em qualquer linguagem de programação existente.

Entretanto, o aprendizado de lógica de programação não é simples e, muitas vezes, é uma barreira para que os estudantes concluam seus cursos. Conforme Ascencio e Campos (2012), um paradigma da programação está diretamente associado com a forma de pensar do programador e a maneira como ele busca solucionar os problemas. Para os autores o limitador

da utilização de alguma técnica de programação é o paradigma e, por meio dele, é possível determinar a maneira que o programador analisou e compreendeu o problema.

### **2.1.2 Aprendizagem colaborativa**

Progressivamente o modelo tradicional de aprendizagem que consiste na frequência dos estudantes em sala de aula para que o professor possa transmitir seu conhecimento passará por transformações. Para Moran, Masetto e Behrens (2015) o paradigma antigo baseava-se na transmissão de conhecimentos do professor e, conseqüentemente, na memorização dos alunos caracterizando uma aprendizagem individualista e competitiva. Para a autora, mesmo que os estudantes saibam as informações ainda não conseguem aplicá-las em situações reais.

Cabe então, aos professores o desafio de mudar o eixo de ensinar para que sejam construídos caminhos que conduzam ao aprendizado. Para interromper o conservadorismo, o professor deve considerar que, além da linguagem oral e da linguagem escrita que norteiam o processo pedagógico de ensinar e aprender, é imprescindível considerar, também, a linguagem digital (MORAN; MASETTO; BEHRENS, 2015).

Neste novo contexto, os estudantes passam a ser os protagonistas na produção do conhecimento. A qualidade desta produção está diretamente ligada ao talento individual de cada aluno e aqueles que tiverem maior facilidade de aprendizagem podem favorecer a produção do conhecimento de outros estudantes. Segundo Torres e Irala (2014), a aprendizagem colaborativa é caracterizada como um conjunto de orientações pedagógicas e bases teóricas propagadas no decorrer da história no contexto escolar.

Para Rangel (2010), os métodos de ensino de aprendizagem colaborativa são produzidos baseando-se em princípios e diretrizes recorrentes ao diálogo, à interação e cooperação entre os estudantes. O autor afirma, ainda, que esta forma de aprendizagem favorecem o intercâmbio de ideias, à discussões e trocas de conhecimento.

Johnson et al. (1990 apud LAAL; LAAL, 2011) destaca cinco elementos principais que compõe a aprendizagem colaborativa e destaca que ela não pode ser considerada simplesmente um sinônimo para estudantes que trabalham em grupos. Um exercício de aprendizagem



somente é caracterizado como aprendizagem colaborativa quando os elementos destacados no Quadro 1 estão presentes.

Quadro 1 - Elementos que compõem a aprendizagem colaborativa

Elemento	Descrição
Interdependência positiva claramente percebida	Os membros da equipe são obrigados a confiar um no outro para atingir o objetivo. Se algum membro da equipe não fizer sua parte, todos sofrerão as consequências. É preciso acreditar que eles estão ligados a outros de uma forma que garanta que todos tenham sucesso juntos.
Interação	Os membros ajudam e incentivam uns aos outros a aprender. Eles fazem isso explicando o que eles entendem, compartilhando conhecimento. Os membros do grupo devem fornecer feedback um ao outro, criticando a opinião dos colegas, além de ensinar e encorajar um ao outro.
Responsabilidade individual e responsabilidade pessoal	Todos os alunos de um grupo são responsáveis por fazer a sua parte do trabalho, mas devem ter domínio de todo o assunto que está sendo estudado.
Habilidades sociais	Os alunos são encorajados e ajudados a desenvolver e praticar a construção de confiança, liderança, habilidades de tomada de decisão, comunicação e gerenciamento de conflitos.
Autoavaliação do grupo	Os membros da equipe definem as metas do grupo, avaliam periodicamente o que estão fazendo bem em equipe e identificam as mudanças que eles farão para funcionar de forma mais eficaz no futuro.

Fonte: Adaptado de Johnson et al. (1990 apud LAAL; LAAL, 2012)

Munhoz (2012) afirma que quando os estudantes trabalham em grupos estão envolvidos na solução de um determinado problema e têm suas formas de desenvolver o conhecimento atendidas, sentem-se inseridas ao ambiente e tendem a aprender melhor os assuntos do que em ambientes que não possuem estas características.

### 2.1.3 Trabalho em grupos de colaboração

Scherer (2005) define a postura interpessoal como uma categoria de estados afetivos que contempla as atitudes de determinada pessoa ao se direcionar a outra, ou seja, o modo como esta pessoa se posiciona afetivamente em comparação aos demais no decorrer de uma interação. Essa postura pode ser desenvolvida de forma espontânea ou pode ser aplicada

estrategicamente no processo de interação com uma ou mais pessoas.

As relações interpessoais são desenvolvidas através do processo de interação. Em situações de trabalhos em grupo há atividades preestabelecidas que devem ser executadas e, para que isso ocorra da melhor maneira possível, é necessário que haja comunicação, cooperação e respeito entre os membros (MOSCOVICI, 2013).

Gahagan (1975) define um grupo como um sistema do qual partes se relacionam entre si e suas metas exercem influências construtivas sobre as atividades de seus membros. Para Moscovici (2013) quando são formados grupos de trabalho, existe uma base interna de diferenças como o conhecimento, informações, preconceitos, atitudes, valores, gostos, fazendo com que exista uma inevitável diversidade de percepções, opiniões e sentimentos em comparação aos demais. A forma com que essas diferenças são tratadas determina como será o relacionamento entre os membros do grupo. Segundo a autora, a forma com que essas diferenças são tratadas possui grande influência sobre o grupo, principalmente nos processos de comunicação e no processo de relacionamento interpessoal.

O relacionamento interpessoal pode tornar-se e manter-se harmonioso e prazeroso, permitindo trabalho cooperativo, em equipe, com integração de esforços, conjugando energias, conhecimentos e experiências para um produto maior que a soma das partes (MOSCOVICI, 2013, p. 70).

Para Gahagan (1975) os grupos colaborativos são considerados por observadores como grupos com maior coordenação, metódicos e produtivos do que grupos onde seus membros são mais individualistas e menos participativos. Entretanto, a autora alerta que o sucesso de um grupo colaborativo resulta sempre em um resultado positivo, pois depende do grau em que a tarefa necessita de atividades coordenadas.

## **2.2 Dimensão tecnológica**

O uso das ferramentas tecnológicas em diversos setores da sociedade é uma necessidade humana de aprimoramento e cada vez mais estão assumindo um papel indispensável ao nosso modelo de vida. Compreende-se por tecnologias educacionais aplicadas ao processo ensino e

aprendizagem não só os recursos disponíveis para auxiliar na produção de conhecimento, mas o próprio conhecimento gerado através dos mesmos.

### **2.2.1 Ambiente colaborativo**

O uso das Tecnologias da Informação e Comunicação (TIC) viabiliza configurar novos ambientes sócio-culturais apoiados por novos modelos de comunicação e processos de estruturação do conhecimento trazendo novos aliados para os ambientes de estudo.

O ambiente colaborativo é uma TIC que vem, ao longo dos anos, ganhando mais espaço no ambiente acadêmico. Henri e Basque (2003) reforçam esse argumento afirmando que a colaboração através de ambientes virtuais está tornando-se uma realidade cada vez mais presente nos dias atuais e pode ser considerada não somente como um método pedagógico, mas como um estilo de vida.

Segundo Seleme e Munhoz (2011) existe uma confusão entre o conceito de ambiente colaborativo e de educação à distância e afirma que o primeiro trata-se de uma ferramenta que pode ser utilizada para a concretização da aprendizagem. Valentini e Soares (2010) definem o ambiente colaborativo como um ambiente voltado para a aprendizagem de algum domínio específico, onde são oferecidas ferramentas para a comunicação. Para os autores, estas ferramentas estão bem lapidadas e manifestam facilidade para o uso e apoio na aprendizagem.

### **2.2.3 Mineração de textos**

Feldman e Sanger (2007) classificam a mineração de textos como uma nova área de pesquisa da ciência da computação que objetiva solucionar a crise da sobrecarga de informação com base em técnicas de mineração de dados, de aprendizagem de máquina, de processamento da linguagem natural, da recuperação de informações e da gestão do conhecimento.

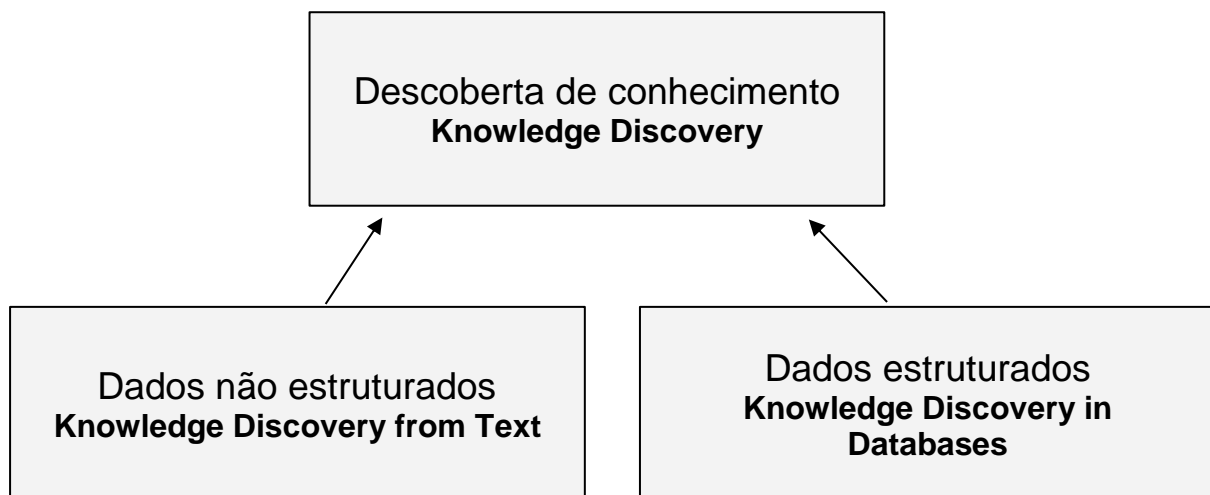
O processamento de linguagem natural (PLN) é uma área de pesquisa com origem no ano de 1950 e é uma das áreas da inteligência artificial cuja preocupação principal é estabelecer a

ligação entre computadores e a linguagem natural falada por pessoas. O PLN é considerado, segundo Lisboa (2013) como uma área de pesquisa independente, entretanto em problemas mais práticos seu uso não é muito recomendado.

Para entender uma consulta em linguagem natural o computador precisa ter conhecimento para analisar e interpretar a entrada de informação. Isso pode significar conhecimento linguístico de palavras, conhecimento sobre áreas específicas, conhecimentos gerais e até menos conhecimento sobre os usuários e seus objetivos. No momento em que o computador entende a informação ele pode agir da forma desejada (TURBAN; ET AL, 2010, p. 168).

Segundo Moraes e Ambrósio (2007), a descoberta de conhecimento apoiada por computador é um processo que analisa dados ou informações com o objetivo de proporcionar novos conhecimentos para a sociedade através da manipulação de uma grande quantidade de dados. Para os autores a descoberta do conhecimento pode ser dividida em duas áreas: Descoberta de conhecimento em dados estruturados (*Knowledge Discovery in Databases-KDD*) e a Descoberta de conhecimento em dados não estruturados (*Knowledge Discovery fromText - KDT*) conforme pode ser visto na Figura 2.

Figura 2 - Diagrama das áreas de descoberta do conhecimento



Fonte: Adaptado de Moraes e Ambrósio (2007, p.2)

A mineração de textos é uma forma de analisar os textos em fóruns de discussão. Dringus e Ellis (2005) realizaram um estudo que avalia o uso da mineração de dados para essa finalidade. Os autores recomendam indicadores de participação onde o professor é capaz de extrair das conversas e, ainda, avaliar a evolução e o desempenho dos seus alunos. Para eles, a

mineração de textos deve ser utilizada para encontrar a distribuição e a frequência que são realizadas contribuições dos alunos nos fóruns. A mineração de textos também pode identificar iniciativas ou respostas e quanto tempo levou para um aluno responder algum questionamento no fórum.

A pesquisa indica que esta técnica possa ser aplicada para avaliar se uma contribuição está associada ao tópico do fórum, e se os alunos compartilham entre eles recursos, como referências retiradas da internet ou citações da literatura (DRINGUS; ELLIS, 2005). Para os autores, a técnica de mineração de textos pode ser uma forma de auxiliar no trabalho do professor em analisar os fóruns de discussão e em obter dados que auxiliem na tomada de decisões para auxiliar os alunos no aprendizado.

Segundo Schenker (2003) na mineração de texto é utilizada com frequência uma técnica para identificar as características de um documento através de um modelo de espaço vetorial onde cada termo torna-se uma característica dimensional. Através de seu valor é possível indicar a quantidade de vezes que um termo aparece no texto ou a quantidade de materiais nos quais o termo é utilizado. Entretanto, o autor ressalta que nessa técnica são descartadas informações importantes como a ordem em que as palavras aparecem, onde aparecem e a sua proximidade.

O conceito de grafo pode ser classificado como uma construção matemática efetiva para desenvolver a modelagem de relacionamento e de informações estruturais e podem ser utilizados para solucionar problema de ordenação, compressão, alocação de recursos, entre outros. Conforme Azevedo, Behar e Reategui (2010) a mineração de textos que utiliza grafos reconhece as palavras que apresentaram maior ocorrência no texto e é capaz de identificar a proximidade entre elas. Segundo os autores, o grafo obtido por meio da mineração de textos exhibe em seus nós as palavras com maior ocorrência e as associações entre os nós retratam a proximidade entre os termos.

Ainda, segundo os autores, por meio dos resultados obtidos pela técnica, o professor poderá realizar um acompanhamento qualitativo das contribuições dos alunos e de um grupo de alunos no fórum de discussão. Assim, o professor será capaz de:

- a) Analisar contribuições textuais que necessitam de intervenção.

- b) Identificar os alunos que realizaram poucas contribuições pertinentes no fórum para oferecer-lhes um maior auxílio.
- c) Incentivar e motivar os alunos que realizaram muitas contribuições relevantes para continuarem interagindo com os colegas que tiveram um menor número de contribuições.

#### **2.2.4 Algoritmos de mineração de textos**

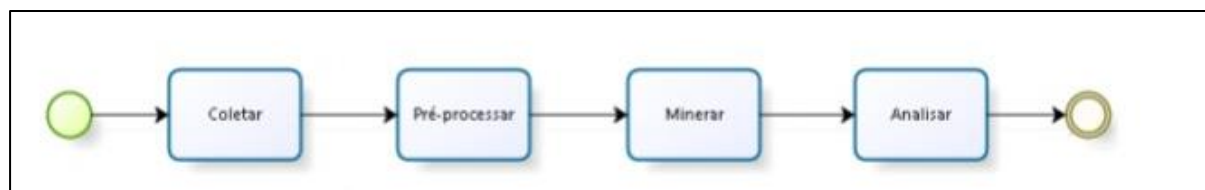
Existem diversos algoritmos para realizar a mineração de textos, entretanto, não existe uma solução única para todos os problemas de inteligência artificial. Da mesma forma, nem sempre uma única técnica pode ser considerada a resposta para todos os problemas relacionados a aprendizagem, classificação e representação de conhecimento (WITTEN; FRANK; HALL, 2011).

Na grande variedade de conjuntos de dados existentes, encontramos diversos tipos de estruturas e um algoritmo de mineração de textos pode em determinado ambiente demonstrar resultados satisfatórios e em outro ser totalmente inadequada (LAHIRI, 2016). Sendo assim, é importante analisar e ter clareza do problema a ser resolvido para encontrar o algoritmo que melhor irá se comportar.

#### **2.2.5 Etapas da mineração de textos**

Segundo Rezende, Marcacini e Moura (2011), as etapas do processo de mineração de textos podem ser classificadas em 4 etapas distintas sendo elas: extração dos dados, pré-processamento, obtenção de conhecimento e interpretação dos resultados obtidos. A Figura 3 descreve este método.

Figura 3 – Processo de descoberta de conhecimento em textos



Fonte: adaptado de Rezende, Marcacini e Moura (2011)

Ainda, para os autores o processo de extração e de organização dos conhecimentos obtidos através da mineração de textos pode ser separado em três etapas principais: pré-processamento dos dados, extração de padrões com agrupamentos de dados e avaliação do conhecimento. O Quadro 2 detalha o processo de descoberta de conhecimento em textos.

Quadro 2 – Detalhamento do processo de descoberta de conhecimento em textos

<b>Etapa</b>	<b>Descrição</b>
Coletar	Esta etapa é muito importante, pois ela é responsável pelo processo de extração dos textos que servirão como base para a análise (SANTOS, 2016). Para Ceci (2010), nesta etapa são selecionados os textos conforme a necessidade do problema, considerando os objetivos que deseja alcançar com o processo.
Pré-processar	Consiste na filtragem e limpeza dos textos, descartando redundâncias e informações irrelevantes para o conhecimento que se deseja extrair. Para Braga (2017) o pré-processamento é importante para identificar grupos semelhantes em um conjunto de textos.
Minerar	É a utilização de um conjunto de métodos e técnicas para realizar a extração do conhecimento de dados não estruturados.
Analisar	É a etapa responsável por validar a eficiência do processo de modo geral, analisando os dados extraídos após a aplicação e utilização dos algoritmos de mineração de textos.

Fonte: da autora (2020)

Neste capítulo foram abordados os temas principais que norteiam o desenvolvimento deste trabalho. A seguir, serão apresentados alguns trabalhos relacionados para trazer comparações ao tema proposto, bem como as dificuldades enfrentadas e resultados obtidos nas pesquisas desenvolvidas.

### **3 TRABALHOS RELACIONADOS**

A seguir serão apresentados trabalhos relacionados à utilização de uma ferramenta colaborativa como apoio a aprendizagem de disciplinas de programação e de métodos que possibilitam a validação da eficácia desta ferramenta através da mineração de textos.

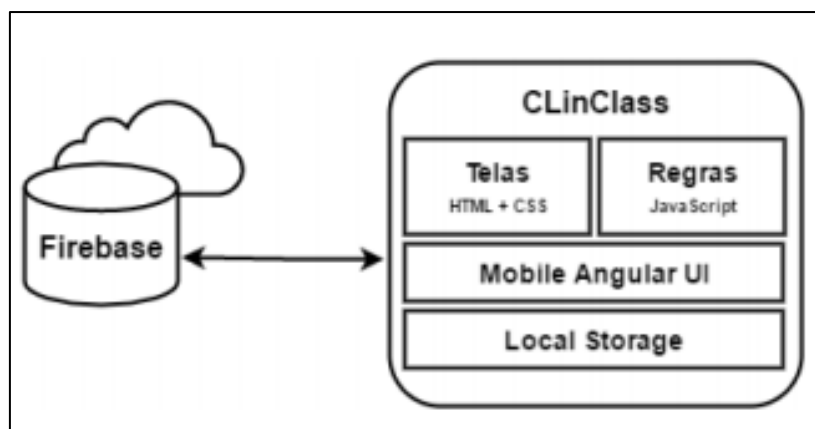
#### **3.1 Ferramenta colaborativa para apoiar a aprendizagem de programação de computadores**

O estudo de Machado *et al.* (2018) aborda a dificuldade da aprendizagem de disciplinas de programação fazendo com que o índice de reprovação e desistência dessas disciplinas seja alto. Buscando uma alternativa capaz de contribuir para o aprendizado através de uma ferramenta colaborativa para dispositivos móveis.

A arquitetura da ferramenta, conforme Figura 4, foi pensada e projetada para permitir que o aplicativo seja utilizado por diferentes sistemas operacionais, mantendo o padrão visual e sua interação com o usuário.



Figura 4 - Arquitetura da ferramenta



Fonte: Machado et al. (2018, p. 24)

Nesta ferramenta foram adotadas duas estratégias de uso por meio da aprendizagem colaborativa. Em um primeiro momento a atividade é realizada de forma individual e, posteriormente, é realizada uma atividade em grupo com a finalidade de realizar o intercâmbio de experiências para aprender a Programação Orientada à Objetos (POO). Assim, na visão do professor o aplicativo possibilita a supervisão de atividades executadas pelo aluno, assim como também é possível visualizar as respostas individuais e dos grupos (MACHADO et al., 2018)

Um elemento importante para a colaboração é a comunicação. Por esse motivo o aplicativo permite, por meio da troca de mensagens de texto, a comunicação entre os alunos e professor, possibilitando que eles esclareçam dúvidas com os pares ou solicitem orientações ao professor. (MACHADO et al., 2018, p.25)

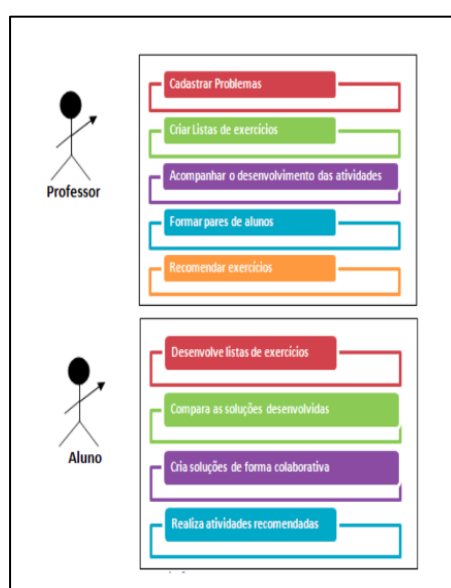
Como resultado do estudo de caso, os autores destacam que o uso da tecnologia para dispositivos móveis, juntamente aplicada com o método de abordagem colaborativa, apresenta benefícios para o aprendizado. Segundo Machado et al. (2018) esta ferramenta incentivou a interação dos alunos nas atividades realizadas e nos trabalhos em grupo, o que resultou em uma redução significativa nos índices de reprovação na turma.

### 3.2 APP- Uma arquitetura pedagógica para aprendizagem de programação de computadores

Em seu artigo Chagas, Oliveira e Tavares (2016) apresentam uma ferramenta pedagógica capaz de apoiar o processo de aprendizagem de linguagens de programação. Para os autores o professor é considerado como um ator do processo de ensino e do processo de aprendizagem. Ele precisa ter ciência de seu papel e deve perceber quando seus alunos não conseguem assimilar os conteúdos abordados. O professor deve refletir sobre sua didática, criticar e ajustar continuamente sua abordagem pedagógica propiciando condições favoráveis à aprendizagem de seus alunos.

Sendo assim, foi criada uma arquitetura pedagógica que promove a aprendizagem de seus usuários e que tem por objetivo melhorar o processo de ensino e do processo de aprendizagem. Através desta arquitetura pedagógica, apresentada na Figura 5, o professor é capaz de avaliar os níveis de aprendizagem de seus alunos de forma contínua. É possível incentivar o desenvolvimento das habilidades através de estratégias de avaliações diagnósticas e da apresentação de atividades que propiciem a autoria, colaboração e o espírito crítico dos alunos.

Figura 5 - Estrutura computacional do APP



Os principais questionamentos levantados neste trabalho referente a eficácia da ferramenta foram respondidos pelos autores, tendo sido apresentadas ao final do trabalho arquiteturas pedagógicas que podem ser aliadas à aprendizagem de programação. Os resultados obtidos foram satisfatórios e indicam que a utilização de uma ferramenta de apoio a aprendizagem pode facilitar a compreensão de assuntos. Chagas, Oliveira e Tavares (2016) esperam que novas pesquisas possam ser desenvolvidas através deste trabalho e anseiam que este estudo possa promover mudanças positivas na aprendizagem de programação.

### **3.3 Comunicação, interação e aprendizagem: o fórum de discussão como estratégia de ensino**

Para Kraemer (2015) o avanço da internet e a constante evolução das formas de comunicação causam grande impacto no ambiente de ensino. O autor afirma que as instituições de ensino estão encarando um dilema, levando em consideração que o modo de ensinar, o papel do professor e a própria sala de aula passaram por pequenas mudanças ao longo do tempo.

Como uma forma alternativa de aprendizagem o autor evidencia a relevância que os fóruns de discussão estão ganhando. Em seu trabalho, Kraemer buscou responder se à interação em fóruns de discussão permitem que seja estabelecido um novo método de aprendizagem.

O trabalho do autor consistiu na análise dos materiais elaborados nas atividades aplicadas em diferentes momentos de disciplinas da Universidade do Vale do Taquari - UNIVATES desenvolvidas em um fórum de discussão. Para realizar esta análise Kraemer realizou a aplicação de questionários aos estudantes, entrevistas e observações das atividades em sala de aula.

Segundo Kraemer (2015), um dos limitadores deste trabalho foi a sua execução somente em um curso de nível superior na área de Ciências Humanas. Para o autor, por ser uma abordagem limitada surge uma série de questões que podem ser utilizadas para realizar novas pesquisas. No entanto, o autor considera que os objetivos da pesquisa foram atingidos e que os fóruns de discussão podem estabelecer uma melhor comunicação entre os alunos.

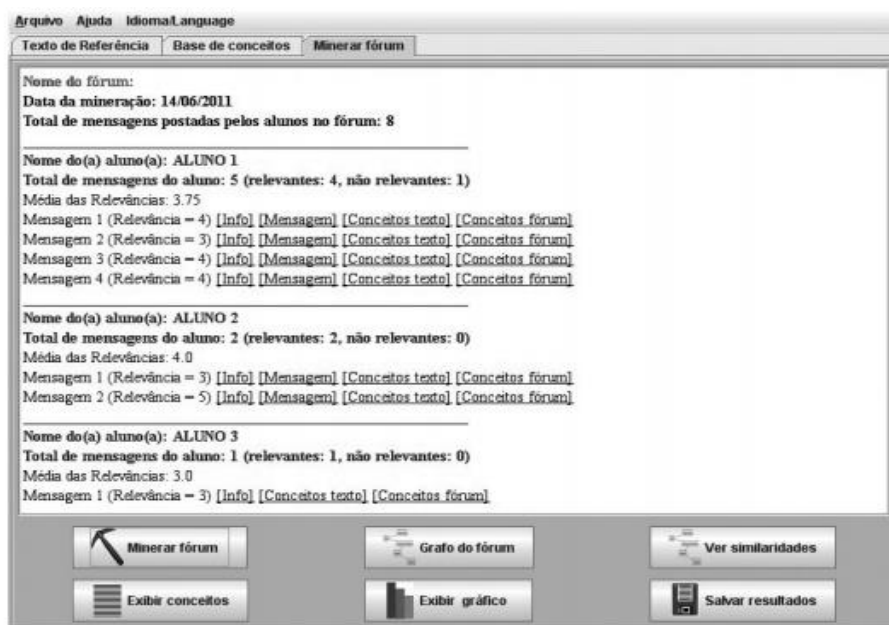
### 3.4 Análise das mensagens de fóruns de discussão através de um software para mineração de textos

Para Azevedo, Behar e Reategui (2011) a socialização em fóruns de discussão pode ser uma atividade importante para o aprendizado dos estudantes. Por meio da análise das interações nos fóruns o professor é capaz de diagnosticar informações sobre o perfil de seus estudantes.

O que motivou o trabalho dos autores é a grande quantidade de alunos e a dificuldade dos professores em analisar todas as discussões realizadas nos fóruns. Segundo os autores uma forma de auxiliar o professor é através de um recurso que permita realizar, de forma automatizada e rápida, a análise de mensagens publicadas em um fórum de discussão. Através desta análise será possível que o professor identifique os alunos que estão interagindo e contribuindo nas discussões.

Para realizar a análise qualitativa dos textos, os autores utilizaram o *software* MineraFórum que tem como base técnicas de mineração de textos para analisar as mensagens enviadas nos fóruns de discussão. Na Figura 6 é apresentada a *Interface* da ferramenta após o usuário selecionar a opção de minerar os dados do fórum.

Figura 6 - Interface do MineraFórum



Após essa análise, foi possível determinar se as contribuições textuais produzidas pelos estudantes são relevantes para o assunto discutido e o professor utilizou critérios próprios. Para analisar as mensagens o software utilizou três critérios: a relevância temática da contribuição textual, a relevância de citações da mensagem e a similaridade da mensagem.

Segundo os autores, os objetivos deste trabalho foram atingidos e o software é capaz de apresentar para o professor uma visão geral das interações entre os seus alunos. A partir das informações coletadas pelo MineraFórum o professor será capaz de direcionar de forma mais adequada seu apoio aos alunos que necessitam um suporte maior para o aprendizado.

### **3.5 Aplicação da mineração de textos em fóruns de discussão**

Azevedo, Behar e Reategui (2010) realizaram uma pesquisa com a técnica de mineração de textos utilizando grafos. Nesta pesquisa, buscou-se realizar uma análise qualitativa das interações textuais realizadas por estudantes em fóruns de discussão. A técnica foi utilizar a mineração para identificar quais estudantes enviaram mensagens que contemplam conceitos pertinentes ao tema da discussão proposta, e quais não o fizeram (AZEVEDO; BEHAR; REATEGUI, 2010).

Para os autores direcionarem as atividades do trabalho foram definidos níveis de relevância temática de uma contribuição textual. Com isso, pretende-se analisar o quanto uma mensagem enviada é relevante para o assunto discutido.

Foram feitas experiências piloto com o objetivo de validar se a técnica de mineração de dados é adequada para analisar as mensagens do fórum de discussão. Através desta análise, o professor obtém dados que podem auxiliá-lo a monitorar as mensagens trocadas pelos estudantes. Como resultado, o autor afirma que o professor é capaz de analisar as contribuições textuais e, também, aquelas que necessitam de intervenção. Através deste trabalho, foi possível identificar os alunos com maior grau de dificuldade de aprendizagem e possibilitou ao professor motivar estes estudantes.

Após a pesquisa e o estudo dos trabalhos relacionados foi possível observar que a utilização de ferramentas de apoio ao aprendizado está ganhando cada vez mais espaço em sala

de aula. Percebe-se que sua utilização qualifica o processo de aprendizagem, o que reforça a hipótese levantada neste trabalho. Também foi possível perceber que a utilização da mineração de textos como forma de avaliar o desempenho e o aprendizado de estudantes pode ser extremamente eficaz, tendo em vista que os resultados das análises são provenientes de uma ampla busca por documentos e sentenças analisadas previamente.

Nos trabalhos analisados não foram destacadas dificuldades que limitassem a pesquisa e avaliação das soluções propostas, entretanto vale ressaltar que ferramentas de apoio ao aprendizado ainda são pouco utilizadas em sala de aula. Como a amostragem é pequena, não há maiores limitações ou obstáculos identificados nas pesquisas. De qualquer forma, pesquisas destas naturezas tendem a ganhar espaço e contribuir para a formação de estudantes de linguagens de programação.

No próximo capítulo serão abordados os métodos e as tecnologias que foram adotados para o desenvolvimento da ferramenta a fim de validar a questão proposta.

## **4 MATERIAIS E MÉTODOS**

Neste capítulo serão apresentados os principais elementos utilizados no desenvolvimento do trabalho, que tem como objetivo validar a efetividade da ferramenta colaborativa integrada ao ambiente de desenvolvimento. É imprescindível para o processo de pesquisa científico definir e seguir uma metodologia que detalhe os caminhos que devem ser percorridos para a execução do projeto. Nesta etapa é compreendida a conceituação teórica do tipo de pesquisa, as ferramentas e tecnologias utilizadas para desenvolver a ferramenta colaborativa e, por fim, será apresentado o desenvolvimento da ferramenta.

### **4.1 Metodologia de pesquisa**

Este trabalho buscou avaliar a eficácia de uma ferramenta para comunicação entre estudantes, integrada ao ambiente de desenvolvimento facilita ou não o aprendizado em sala de aula.

Segundo Gil (2008) o tipo de uma pesquisa pode ser classificado quanto à sua natureza, abordagem do problema, objetivos e procedimentos metodológicos. Para alcançar os resultados esperados neste trabalho foi inevitável realizar uma pesquisa exploratória dos temas relacionados. Lakatos e Marconi (2010) explicam que o estudo exploratório pode ser realizado por meio de documentos e é formado, basicamente, por três objetivos essenciais: desenvolver hipóteses, ampliar o conhecimento sobre o assunto a ser estudado e afirmar se é fato ou fenômeno. Desta forma, é preciso apresentar uma hipótese de trabalho para atingir os resultados esperados em um nível de melhor interpretação.

A hipótese levantada neste trabalho é baseada na comparação de estudos já realizados, que tem como justificativa o fato de que ter um ambiente de desenvolvimento aproximando os alunos e facilitando a comunicação entre eles possa favorecer o aprendizado de disciplinas de lógica de programação. Neste trabalho foi proposto o desenvolvimento de uma ferramenta de comunicação, juntamente ao ambiente de desenvolvimento utilizado pelos estudantes, com a finalidade de avaliar e medir a eficácia da ferramenta, caracterizando a pesquisa como exploratória.

A pesquisa exploratória pode ser classificada, conforme Gil (2008), como: pesquisa bibliográfica e estudo de caso. Segundo o autor, na pesquisa bibliográfica são buscadas ideias já referenciadas em bibliografias no que diz respeito ao tema da pesquisa e realizando uma análise crítica da situação atual. O estudo de caso é utilizado, normalmente, em casos onde o questionamento da pesquisa está relacionada à determinação de como ou por quê determinado fenômeno ocorre, não havendo a necessidade de monitorar os eventos. É um método utilizado para examinar casos particulares, mas que sejam representativos e favoreçam a elaboração de hipóteses a partir de construções teóricas (SAKAMOTO; SILVEIRA, 2019; YIN, 2015).

A modalidade da pesquisa abordada neste trabalho, conforme procedimentos técnicos, pode ser caracterizada como estudo de caso, que, conforme Gil (2008), é um estudo profundo e, por vezes, exaustivo de objetos, explorando-os de forma ampla e bastante detalhada. Geralmente é utilizada uma amostra reduzida, que tem como limitação não generalizar os resultados. Entretanto, o objetivo do estudo de caso é não caracterizar a amostra e sim estudar o problema do aprendizado de lógica de programação e identificar os fatores ou contribuições da utilização de uma ferramenta integrada ao ambiente de desenvolvimento.

Este trabalho não teve com finalidade analisar um grande volume de dados gerados por múltiplas turmas. Entretanto, para que fosse possível avaliar seu uso foi utilizada a abordagem de pesquisa qualitativa. Prodanov e Freitas (2013) acreditam que a pesquisa qualitativa possui relação direta entre o real e o indivíduo tendo vínculo a qualidade do assunto abordado e não apenas em dados numéricos.

Nesta pesquisa foram observadas duas turmas das disciplinas de Projeto Integrador I e Programação Orientada a Objetos. Os dados foram levantados através da aplicação de questionários com os estudantes e com o professor das disciplinas e, após, foram analisados e utilizados no trabalho como forma de validar a proposta desta pesquisa. Também foi possível

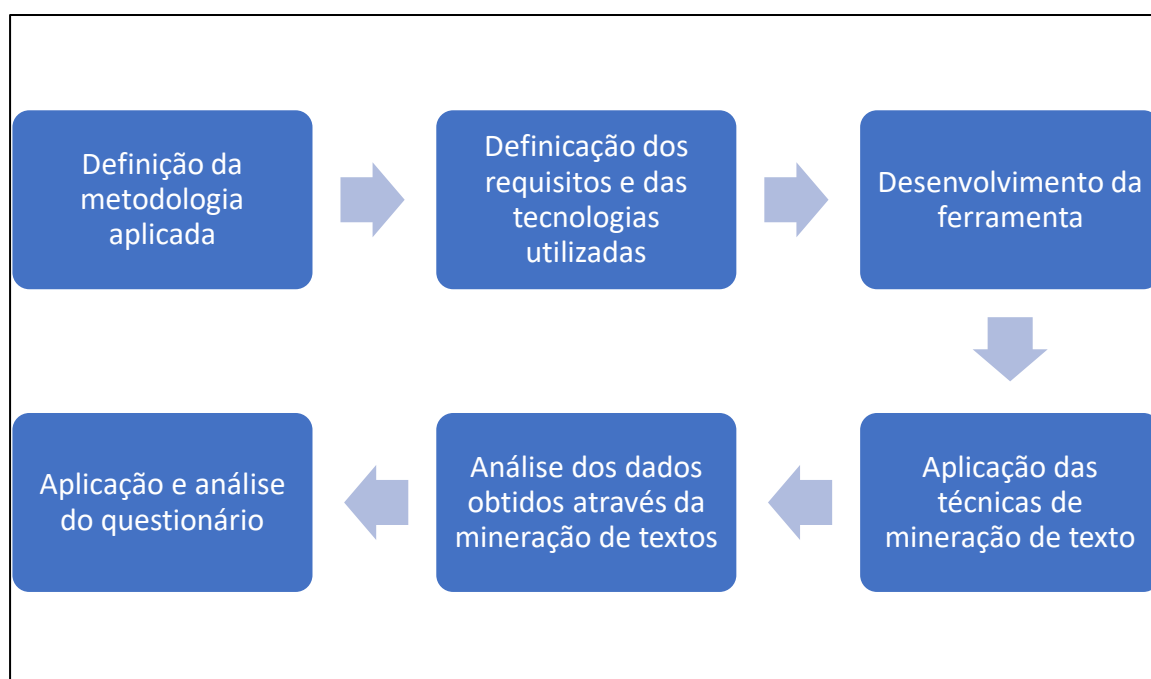


contar com análises processadas pela ferramenta através da análise dos textos e a obtenção de estatísticas relevantes para o professor avaliar a influência da utilização da mesma. Após a análise destes resultados serão apresentadas as conclusões obtidas.

Para ilustrar o processo de desenvolvimento do trabalho em detalhes, foi confeccionado o fluxograma da Figura 7. O primeiro passo foi a definição das metodologias utilizadas para o desenvolvimento da pesquisa e do método utilizado para validar a solução proposta. Na sequência foi definido como seria desenvolvida a ferramenta, bem como as tecnologias que seriam utilizadas e, posteriormente, deu-se início ao seu desenvolvimento.

Para auxiliar a validação da solução proposta foram aplicadas as técnicas de mineração de textos, através de um algoritmo desenvolvido, para que fossem analisados os padrões de comportamento dos estudantes que utilizariam a ferramenta. Após a obtenção dos dados foi realizada a análise da participação dos estudantes como forma de auxiliar os professores a realizar as intervenções necessárias para contribuir no aprendizado. Por fim, como forma de validar a solução proposta foi aplicado e analisado um questionário com os estudantes.

Figura 7 – Fluxograma do desenvolvimento do trabalho



## **4.2 Tecnologias utilizadas**

Nesta seção serão detalhadas as principais ferramentas e tecnologias que serão utilizadas no desenvolvimento do trabalho.

### **4.2.1 Apache**

Conforme Marcelo (2005), o Apache é um servidor HTTP mantido pela fundação Apache Group e é bastante popular entre as aplicações web. É responsável por controlar com segurança as transações entre cliente e servidor. É um software livre e suporta diferentes linguagens de programação, como PHP, Java e outras.

### **4.2.2 HTML e CSS**

O *HyperText Markup Language* (HTML) é a principal linguagem de programação utilizada para desenvolver páginas web. Ela permite desenvolver documentos estruturados em títulos, parágrafos, listas, formulários e em diversos elementos nos quais é possível incorporar imagens, e objetos, como, por exemplo, um vídeo (FLATSCHART, 2011).

O *Cascading Style Sheets* (CSS) é responsável por promover o acabamento visual das páginas web. Ele pode ser compartilhado entre várias páginas, permitindo que haja uma padronização visual simplificada e lógica (BONATTI, 2013).

### **4.2.3 PHP**

Segundo Dall'Oglio (2016), no início o PHP era chamado de *Personal Home Page Tools / Formsinterpreter* e era composto, em sua maioria, por um conjunto de scripts escritos

em linguagem C voltados para criação de páginas dinâmicas. Com o tempo e com o aumento da utilização desta linguagem, foram adicionados novos recursos, como a integração ao banco de dados. O PHP passa, então, a ser chamado de Hypertext Preprocessor.

É uma das linguagens mais utilizadas no mundo e sua popularidade se deve à facilidade em criar aplicações dinâmicas, suportando a maioria dos bancos de dados existentes, e ao conjunto de funções que através de uma estrutura flexível permite desde à criação de simples portais até aplicações mais complexas (DALL’OGLIO, 2016).

#### **4.2.4 Adianti Framework**

O Adianti Framework é uma arquitetura open-source utilizada na criação de sistemas em PHP baseado no padrão de projetos *ModelViewController*(MVC). Este padrão objetiva separar as camadas lógicas da aplicação, deixando o código desenvolvido mais organizado e estruturado (DALL’OGLIO, 2015).

Conforme Dall’Oglio (2015) para facilitar a criação de sistemas, esta arquitetura fornece componentes a construção de formulários e *datagrids*, relatórios, gráficos, entre outros. Dessa forma, o desenvolvedor não possui a necessidade de se preocupar com detalhes de implementação, limitando-se a focar na lógica e na regra de negócios da aplicação.

#### **4.2.5 PostgreSQL**

Milani (2008) define o PostgreSQL como um Sistema Gerenciador de Banco de dados (SGBD) objeto relacional. É responsável por armazenar e gerenciar todas as informações de um sistema de informação. É considerado um SGBD muito confiável, estável e possui código fonte aberto, possibilitando um maior crescimento através da contribuição da comunidade de desenvolvedores (MILANI 2008).

#### 4.2.6 NetBeans IDE

O NetBeans IDE possibilita o desenvolvimento de aplicações desktop Java, móveis e Web, e também aplicações com HTML, JavaScript e CSS. Esta IDE fornece um conjunto de ferramentas para desenvolvedores de PHP e C/C++. Além disso, também fornece modelos de código, dicas de codificação e ferramentas de refatoração. O NetBeans IDE é gratuito e possui código-fonte aberto (NETBEANS, 2020).

#### 4.2.7 Java

O Java é uma linguagem de programação orientada à objetos e para muitos é considerada similar às linguagens C, C++ e C# (JORGE, 2004).

Sua principal característica é que os programas que são criados nessa linguagem não são compilados em código nativo da plataforma, sendo compilados para um *bytecode* executado por uma máquina virtual. Isso permite aos desenvolvedores a criação de um programa uma única vez, podendo ser executado em qualquer uma das plataformas suportadas pela tecnologia (BRITO, 2014).

#### 4.2.8 Web Services

Kuehne (2009) define um *webservice* (WS) como uma implementação da arquitetura *Service Oriented Architecture* (SOA), sendo descrito basicamente como um mecanismo para a comunicação entre aplicações distintas. Duarte (2014) descreve os WS como aplicações modulares, que podem ser escritas e publicadas sem que o solicitante precise se preocupar com a linguagem ou o ambiente em que esta funcionalidade foi implementada.

Para Kuehne (2009) o principal objetivo de um WS é unir diferentes sistemas, tornando as aplicações mais modulares, encapsulando métodos que serão requisitados de forma remota

por algum protocolo de comunicação. O autor destaca que o ciclo de vida de um WS é composto basicamente por quatro estados:

- a) Construção: fase em que são contemplados o desenvolvimento e os testes da funcionalidade;
- b) Publicação: nesta fase é feita a publicação dos serviços desenvolvidos;
- c) Execução: fase após a publicação do WS, onde o consumidor já pode utilizar as funcionalidades;
- d) Gerenciamento: responsável por administrar o serviço criado, respeitando os princípios de segurança, disponibilidade, desempenho e qualidade.

#### 4.2.9 Python

É uma linguagem de programação orientada a objetos interativa e interpretada, possuindo uma sintaxe considerada clara e de fácil compreensão. Pode ser estendida em C ou C++ e como é uma linguagem multiplataforma é possível executá-la em Mac, Unix e Windows (PYTHON, 2019).

Para Carvalho e Branco (2016) Python é uma linguagem ideal para programar, pois proporciona muita flexibilidade na forma de estruturar os programas. Ela pode ser utilizada como linguagem de *script* com a finalidade de executar o código-fonte, como linguagem de procedimentos para organizar programas ou como linguagem orientada a objetos para usar classes, heranças e módulos para montar uma hierarquia. A flexibilidade dessa linguagem permite definir o melhor estilo de programação para o desenvolvimento de projetos.

#### 4.2.10 Bibliotecas Python

As principais bibliotecas do Python utilizadas para desenvolver este trabalho estão listadas a seguir:

- **Numpy:** Biblioteca que fornece suporte a grandes vetores, vetores multidimensionais

e matrizes, além de uma quantidade considerável de funções matemáticas para serem operadas nestes vetores. (VAN DER WALT et. al., 2011)

- **NLTK:** De acordo com Bird, Klein e Loper (2019) o pacote *Natural Language Toolkit* (NLTK) realiza o processamento de linguagem natural em Python. Essa ferramenta é utilizada em aplicações de tratamento de texto, como análise de sentimentos em discursos e mineração de opiniões do público em publicações (KOTHAPALLI; SHARIFAHMADIAN; SHIH, 2016).

#### 4.2.11 Naive Bayes

Criado por Thomas Bayes, um matemático inglês, o Naive Bayes é um algoritmo classificador probabilístico utilizado em aprendizado de máquina para categorizar textos com base na frequência com que as palavras são utilizadas (GOMES, 2019).

Para Tamais (2019) o algoritmo possui uma parte matemática considerada simples, possui um desempenho satisfatório e necessita poucas observações para atingir uma boa acurácia. A tabela de probabilidade do Naive Bayes pode ser visualizada na Figura 8.

Figura 8 – Tabela de probabilidade Naive Bayes

Y		X	
		0 número de obs com X = 0	1 número de obs com X = 1
0	$P(Y=0)$	$P((X = 0) \mid (Y = 0))$	$P((X=1) \mid (Y=0))$
1	$P(Y=1)$	$P((X = 0) \mid (Y = 1))$	$P((X=1) \mid (Y=1))$

Fonte: adaptado de Tamais (2019)

Como ilustrado na Figura 9, para realizar a classificação o algoritmo considera as *features* independentes duas a duas e pode-se concluir que a probabilidade de  $X_i$  é a mesma probabilidade de  $X_i$  dado  $y$ , pois o valor das outras *features* é irrelevante para a probabilidade da *feature* em questão.

Figura 9 – Probabilidade do Naive Bayes

$$P(x_i | y, x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_n) = P(x_i | y)$$

Fonte: adaptado de Tamais (2019)

Desta forma, pode-se considerar que  $P(x_1, \dots, x_n | y) = P(x_1 | y) * P(x_2 | y) * \dots * P(x_n | y)$ . Como  $P(x_1, \dots, x_n)$  é uma constante, pode-se considerar que  $P(y | x_1, \dots, x_n) \propto P(y) \prod P(x_i | y)$  e o classificador escolhe o  $y$  cuja a probabilidade é maior conforme Figura 10.

Figura 10 – Classificador de Naive Bayes

$$\hat{y} = \arg \max_y P(y) \prod_{i=1}^n P(x_i | y)$$

Fonte: adaptado de Tamais (2019)

### 4.3 Desenvolvimento do projeto

Nesta seção é detalhada a solução em software que será desenvolvida para auxiliar a colaboração em grupos de forma integrada ao ambiente de programação, facilitando a comunicação e especialmente a aprendizagem de lógica de programação nos períodos iniciais dos cursos de TI. Sendo assim, a seguir será apresentada a documentação da ferramenta, através de seus requisitos, do modelo UML de casos de usos, do modelo entidade relacionamento (ER), do modelo do banco de dados e sua interface.

### 4.3.1 Requisitos

Existem, basicamente, dois tipos de requisitos que são diferenciados como funcionais e não funcionais. Os requisitos funcionais são considerados funcionalidades ou serviços essenciais e que o sistema deve contemplar. Já os requisitos não funcionais são considerados uma necessidade funcional, ou seja, são aqueles que definem como as funcionalidades ou serviços são implementados.

#### 4.3.1.1 Requisitos funcionais

Para realizar o desenvolvimento da ferramenta proposta neste trabalho alguns requisitos funcionais devem ser atendidos, conforme es especificados no Quadro 3.

Quadro 3 - Lista de requisitos funcionais

<b>RF01 - Manter o cadastro de instituições</b>	<b>Prioridade</b>
Inserir, excluir ou editar informações sobre as instituições de ensino usuárias do sistema.	Baixa
<b>RF02 - Manter o cadastro de turmas</b>	<b>Prioridade</b>
Inserir, excluir ou editar informações sobre as turmas, vinculadas a uma instituição de ensino, identificando a disciplina e o professor	Alta
<b>RF03 - Manter o cadastro de atividades</b>	<b>Prioridade</b>
Inserir, excluir ou editar informações sobre as atividades/exercícios que os serão realizados pelos grupos de colaboração	Alta
<b>RF04 - Manter o cadastro de pessoas</b>	<b>Prioridade</b>
Inserir, excluir ou editar informações sobre os alunos, professores e administradores do sistema	Alta
<b>RF05 - Manter o cadastro de grupos</b>	<b>Prioridade</b>
Inserir, excluir ou editar grupos de colaboração, vinculados a uma turma, permitindo a vinculação de alunos ao grupo	Alta
<b>RF06 - Enviar mensagem</b>	<b>Prioridade</b>



Permitir que um aluno (membro de um grupo) inicie uma conversa, postando uma mensagem	Alta
<b>RF07 - Receber mensagem</b>	<b>Prioridade</b>
Permitir que um aluno dê continuidade a uma conversa iniciada, postando uma mensagem complementar	Alta
<b>RF08 - Qualificar os grupos</b>	<b>Prioridade</b>
Produzir informações indicativas sobre a postura interpessoal solicitude	Alta
<b>RF09 - Publicar atividades</b>	<b>Prioridade</b>
Disponibilizar aos grupos de colaboração um determinado enunciado de atividade.	Baixa
<b>RF10 - Produzir estatísticas sobre a atividade dos grupos</b>	<b>Prioridade</b>
Produzir outras informações estatísticas para subsidiar decisões e estratégias pedagógicas do docente	Média

Fonte: da autora, 2020.

#### 4.3.1.2 Requisitos não funcionais

A seguir, no Quadro 4, serão listados os requisitos não funcionais que deverão ser implementados para a ferramenta proposta:

Quadro 4 - Lista de requisitos não funcionais

<b>RNF01 - Autenticar usuários</b>	<b>Prioridade</b>
O acesso ao sistema por qualquer um dos usuários deve ser permitido somente via identificação do usuário e verificação de sua credencial por meio de senha	Alta
<b>RNF02 - Estar disponível em uma plataforma web</b>	<b>Prioridade</b>
A interface do professor e do administrador deve ser executada por meio de um navegador web	Alta
<b>RNF03 - Estar disponível ao aluno no mesmo ambiente de produção (IDE) que utiliza para programar</b>	<b>Prioridade</b>
A interface de interação dos grupos de colaboração deve ser incorporada à IDE Netbeans por meio de plugin disponibilizado ao aluno	Alta

<b>RNF04 - Utilizar banco de dados PostgreSQL</b>	<b>Prioridade</b>
O banco de dados PostgreSQL é disponibilizado sob licenciamento que permite sua utilização neste software, além de ser reconhecimento pela comunidade de Software Livre como confiável e robusto	Alta
<b>RNF05 - Permitir a exportação de dados em formato CSV</b>	<b>Prioridade</b>
Os dados como lista de alunos ou grupos, disponibilizados em arquivos no formato CSV, podem ser facilmente acessados e trabalhados pelo docente ou pesquisador em outros softwares	Média
<b>RNF06 - Utilizar PHP 7.1</b>	<b>Prioridade</b>
Desenvolver utilizando PHP versão 7.1 ou superior como linguagem de programação back-end	Alta
<b>RNF07- Utilizar Adianti Framework</b>	<b>Prioridade</b>
Utilizar Adianti Framework no desenvolvimento do back-end	Alta
<b>RNF08 - Compatível com navegador Google Chrome</b>	<b>Prioridade</b>
Ser compatível com navegadores Google Chrome (versão 70 ou superior)	Alta

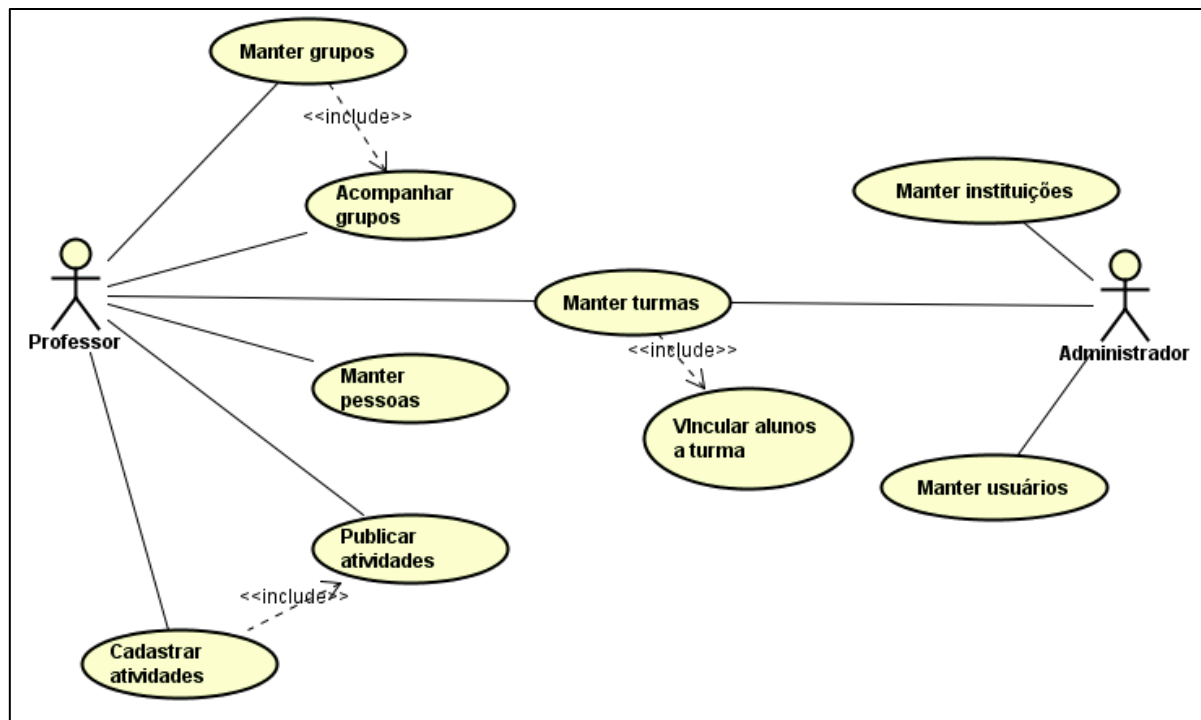
Fonte: da autora, 2020.

#### 4.3.2 Modelo de casos de uso

Para demonstrar os principais papéis dentro da ferramenta proposta será utilizado o diagrama de casos de uso. Conforme visto na Figura 11 e na Figura 12, a ferramenta possui três grupos de usuários: o grupo de professores que é responsável por todos os cadastros e análises que serão realizadas nos dados relacionados às funcionalidades implementadas na ferramenta; o grupo de administradores do sistema que é responsável por administrar e gerenciar os acessos na ferramenta proposta; e o grupo de estudantes que pode realizar interações no fórum de discussão proposto pelo professor.

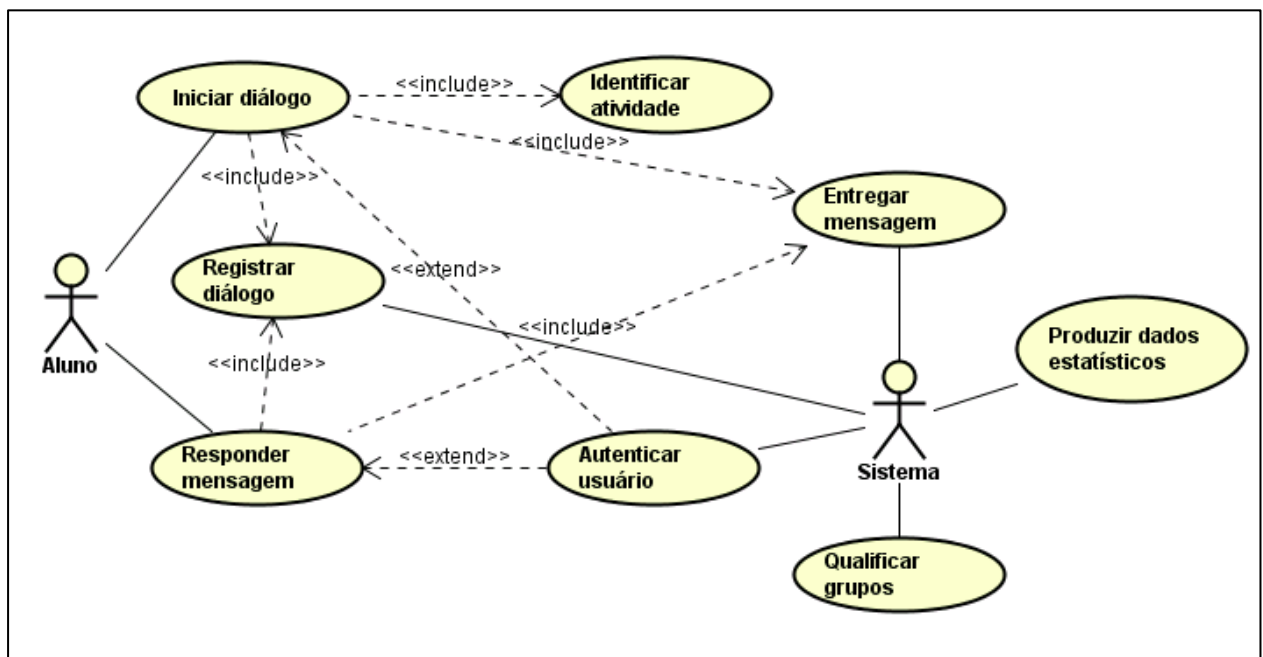
Ainda, além dos grupos descritos acima, podemos considerar que o sistema também terá tarefas na ferramenta, como, por exemplo, gerenciar as requisições do *WebService* entre a ferramenta utilizada pelos estudantes e a plataforma de gerenciamento do professor.

Figura 11 - Modelo de caso de uso da plataforma para professores e administradores



Fonte: da autora, 2020.

Figura 12 - Modelo de caso de uso da plataforma para alunos

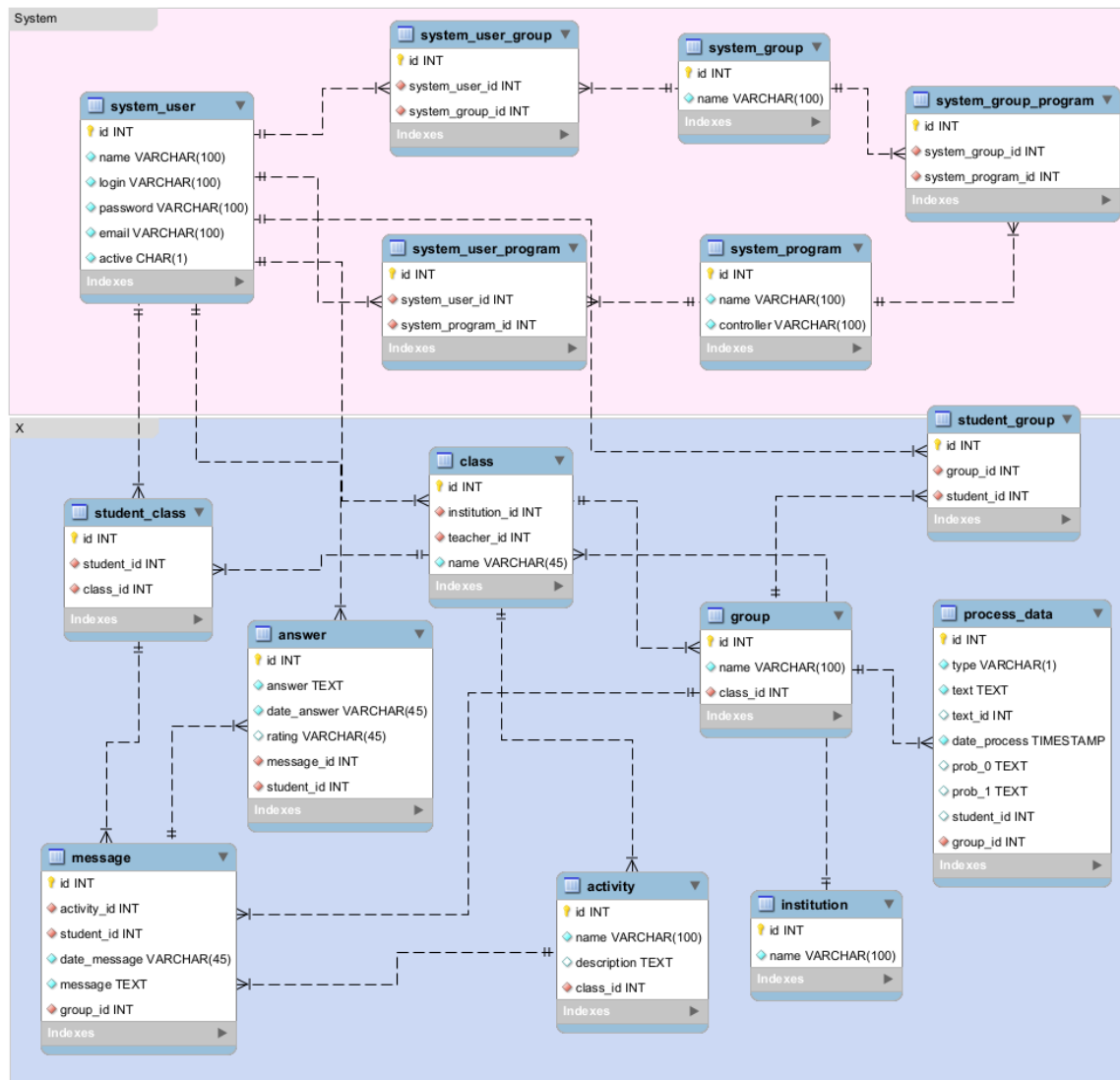


Fonte: da autora, 2020.

### 4.3.3 Modelo relacional

Para representar visualmente como as informações serão armazenadas na aplicação foi elaborado o modelo relacional apresentado na Figura 13.

Figura 13 - Modelo do banco de dados



Fonte: da autora (2020).

O modelo contempla os cadastros básicos de usuários, grupos de usuários e permissões de acesso ao sistema; turmas e instituições que modelam os dados de contexto, ou seja, modelo padrão encontrado nas instituições de ensino. Os alunos podem estar matriculados em várias turmas e para cada turma que estarão poderão interagir através de troca de mensagens com os

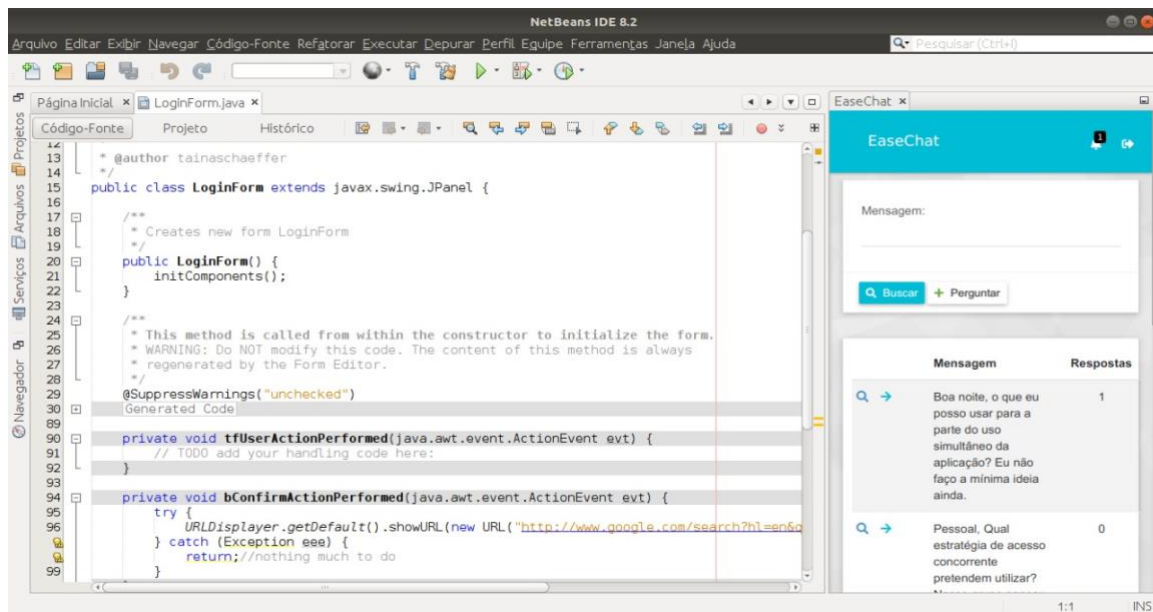
colegas. A tabela “group” armazena as informações sobre os grupos da turma e sobre informações estratégicas sobre eles. As mensagens trocadas pelos alunos do grupo serão armazenadas na tabela “message” e cada entidade desta tabela está relacionada a uma atividade cadastrada pelo professor para a turma. A tabela “process\_data” armazena as informações referentes a análise e processamento dos dados realizada pelo algoritmo de mineração de textos.

#### **4.3.4 Instalação e interface da ferramenta**

Com o objetivo de auxiliar os alunos na instalação e utilização do “EaseChat” foi disponibilizado o manual do Apêndice A, onde constam todas as etapas de *download* e instalação do *plug-in* e o tutorial de uso da ferramenta. A seguir, são detalhadas as principais funcionalidades da ferramenta para o uso no perfil Aluno e no perfil Professor.

A Figura 14 apresenta uma das telas das principais do sistema, ou seja, o ambiente utilizado pelos grupos de colaboração para visualizar as perguntas realizadas por outros colegas de grupo. Nesta tela é possível visualizar dados como o autor da pergunta e quantas respostas determinada pergunta já obteve. Nesta tela também é possível iniciar uma nova pergunta através do botão “Perguntar”.

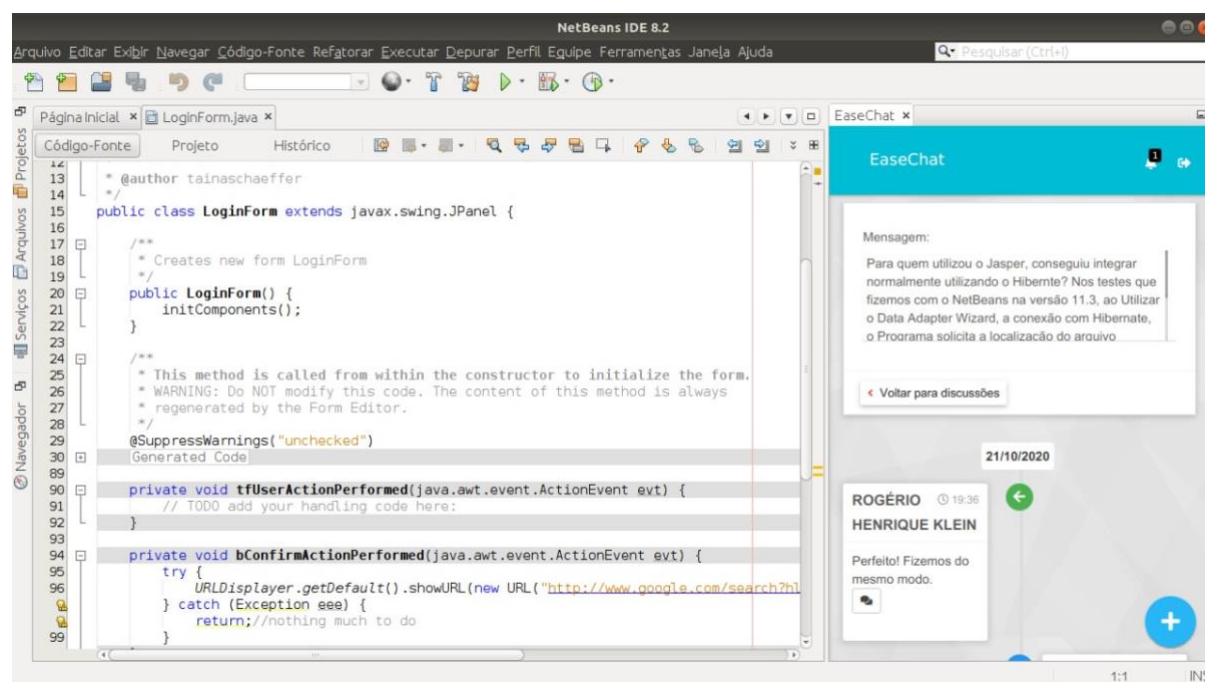
Figura 14 - Tela de listagem de discussões do grupo



Fonte: da autora (2020).

Na Figura 15 é possível ver a tela de discussões de uma pergunta. Nesta tela aparecem todas as mensagens enviadas como resposta para uma dúvida. Para enviar sua resposta o aluno precisa clicar no botão “+” no canto direito inferior do *plug-in*. As mensagens podem ser visualizadas em formato de linha do tempo, onde as respostas mais recentes são exibidas no topo.

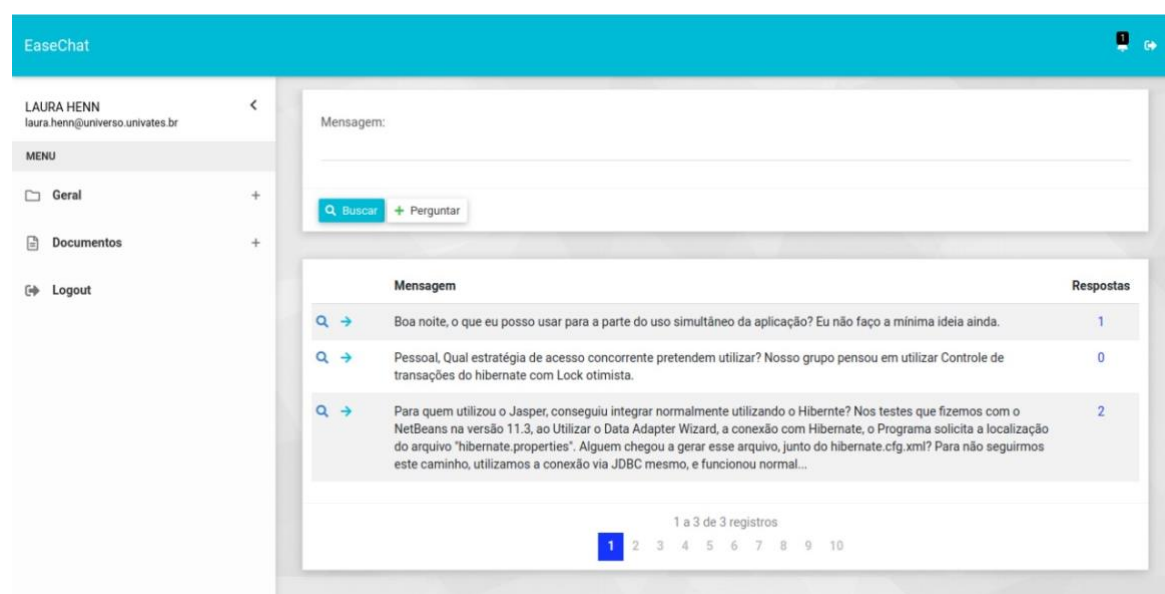
Figura 15 – Tela de discussões



Fonte: da autora (2020)

Para facilitar e aumentar o alcance do uso da ferramenta ela também pode ser utilizada na versão *web*, ou seja, sem que seja necessário instalar e utilizar o *plug-in*, conforme Figura 16. Também foi pensado em disponibilizar a ferramenta na versão *web* para que os alunos possam utilizar o “EaseChat” a qualquer momento, como, por exemplo, pelo celular.

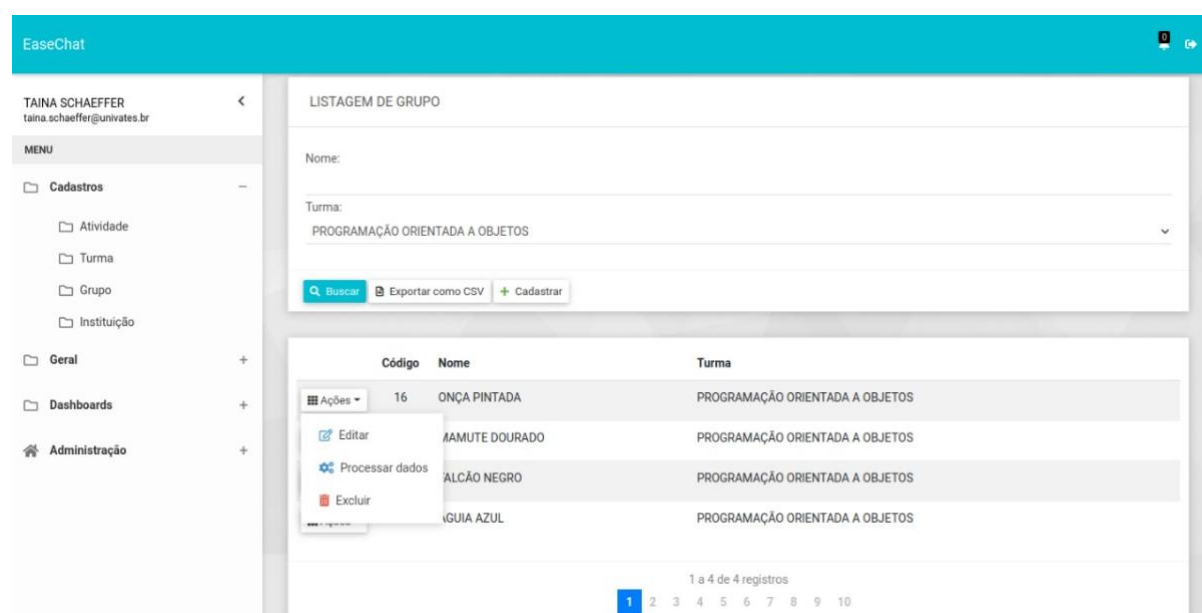
Figura 16 – Visualização da listagem de discussões do EaseChat web



Fonte: da autora (2020)

Para o professor realizar a análise do conteúdo trocado entre os grupos é necessário realizar o processamento dos dados na tela de cadastro dos grupos. Esse recurso coleta todas as perguntas e respostas feitas no grupo e as envia para o algoritmo de mineração de textos responsável pela análise de solicitude. Sendo assim, na tela de listagem dos grupos o professor escolhe o que deseja visualizar os dados e através do botão “Ações” seleciona a opção “Processar dados”. A Figura 17 ilustra este processo.

Figura 17 – Processamento dos dados do grupo no minerador de textos



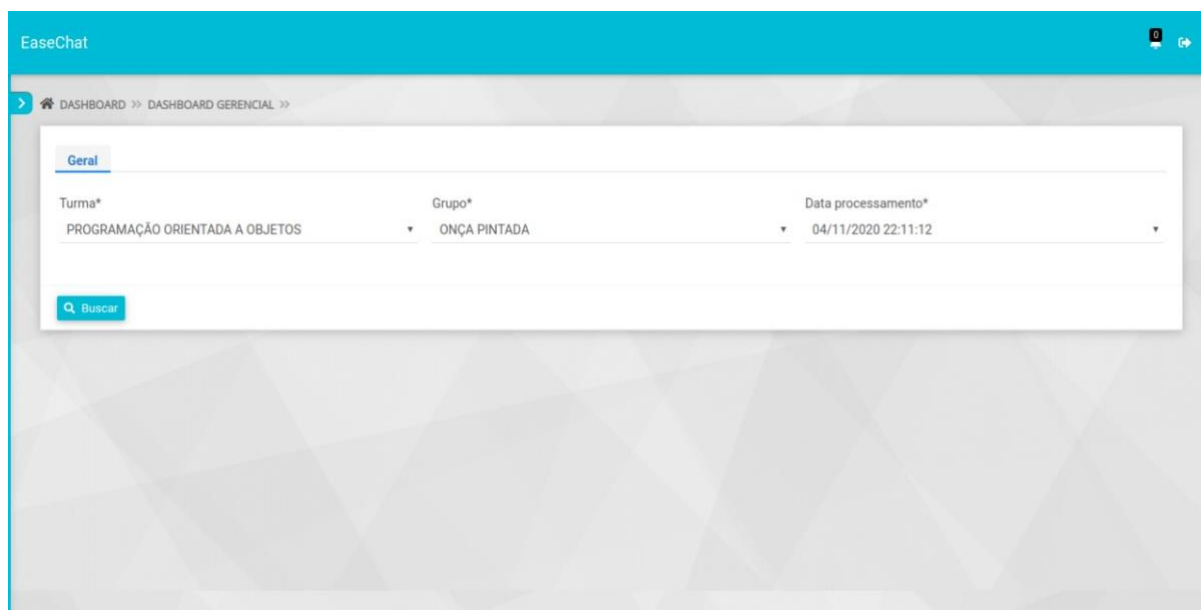
Fonte: da autora (2020)

O resultado obtido com o processamento de dados pode ser acompanhado pelo professor no *Dashboard* gerencial. Esse *Dashboard* tem a finalidade fornecer ao professor recursos para analisar o progresso e desenvolvimento da turma. Esse acompanhamento será em tempo real, fazendo com que o professor fique mais próximo de seus alunos e consiga identificar com maior antecedência quais estudantes estão necessitando de maior suporte.

Os dados são exibidos por grupos, sendo assim o professor deve indicar a turma que deseja visualizar, o grupo e o momento do processamento dos dados pois o professor pode realizá-lo várias vezes durante o andamento das atividades. A Figura 18 mostra como é realizado este processo.



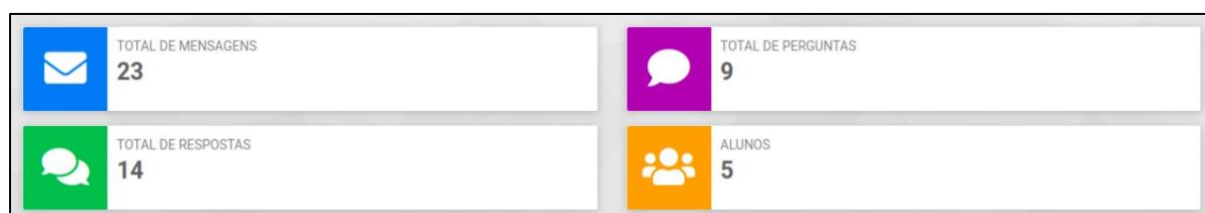
Figura 18 – Busca dos dados do grupo no Dashboard



Fonte: da autora (2020)

Ao carregar o *Dashboard* são exibidos os indicadores para o professor. A Figura 19 mostra os primeiros indicadores exibindo, assim, o total de alunos dentro do grupo, o total perguntas e o total de respostas realizadas pelo grupo, bem como o total de mensagens trocadas. Esses dados podem auxiliar o professor indicando se o grupo é participativo, ou seja, realiza a troca de um número significativo de mensagens, ou não.

Figura 19 – Indicadores do Dashboard



Fonte: da autora (2020)

Na sequência, o *Dashboad* exibe os dados referentes ao processamento no algoritmo de mineração dos dados. Este relatório exibe a quantidade de participações que o aluno teve no grupo, ou seja, o total de perguntas e respostas feitas; o total de participações classificadas pelo algoritmo como solícitas e o percentual que isso representa em relação ao total de participações; o índice geral de solicitude, ou seja, a relação entre o total de participações e o total de

participações solícitas. Ainda neste relatório, a mesma avaliação é feita, porém, considerando apenas as respostas feitas pelos alunos. No final, o relatório exhibe a quantidade de perguntas que o aluno realizou. Com esses indicadores o professor pode avaliar se os integrantes do grupo estão sendo participativos e contribuindo com as dúvidas dos colegas. Este relatório pode ser visualizado na Figura 20.

Figura 20 – Relatório de análise de solicitude

ANÁLISE DE SOLICITUDE							
Exportar							
Aluno	Total de participações	Total de participações solícitas	Índice geral de solicitude	Total de respostas	Total de respostas solícitas	Índice geral de solicitude nas respostas	Total de perguntas interagidas
ARTHUR CORBELLINI	7	3 (42.86%)	13.04%	3	2 (66.67%)	14.29%	3
CRISTIANO SANGALLI FIEL	5	0 (0%)	0%	4	0 (0%)	0%	4
DAIANA MACHADO DA COSTA	2	1 (50%)	4.35%	1	0 (0%)	0%	1
ENZO WEINBERGER MOSMAN	5	1 (20%)	4.35%	3	0 (0%)	0%	3
RAFAEL MICHEL DOCENA	4	3 (75%)	13.04%	3	2 (66.67%)	14.29%	3

Fonte: da autora (2020)

Na Figura 21 é apresentado o último relatório disponibilizado no *Dashboard*. Este relatório exhibe as mensagens das discussões iniciadas pelos alunos do grupo, onde é possível ver informações como a atividade da pergunta, o aluno autor da pergunta, a mensagem que ele publicou e o número total de respostas que a mensagem recebeu. Ao clicar no total de respostas é aberta um detalhamento com as respostas dadas pelos integrantes do grupo.

Figura 21 – Listagem de perguntas

LISTAGEM DE PERGUNTAS						Exportar
Turma	Atividade	Grupo	Aluno	Mensagem	Respostas	
PROGRAMAÇÃO ORIENTADA A OBJETOS	Sprint 8 - Dashboard e e-mail	ONÇA PINTADA	ARTHUR CORBELLINI	Boa noite colegas. Estou com dúvida na questão abaixo e não tenho certeza se a minha resposta está certa. O que vcs acham? Segue - PERGUNTA: Se uma classe tem o método construtor sobrecarregado, como um objeto novo será instanciado, ou seja, qual construtor será usado para produzir um novo objeto? - RESPOSTA: Depende se irá ou não entrar algum atributo dentro do construtor. Para exemplificar, considerando uma classe chamada "Cliente" com os atributos "nome", "idade" e "telefone". Quando há sobrecarga do método construtor, podemos instanciar tal objeto de duas formas: Ao instanciar uma nova classe sem passar nenhum valor de entrada, os atributos deste novo objeto ficam nulos (ou "default"). Já quando instanciamos uma nova classe e passamos valores de entrada, o objeto criado terá tais valores dentro de seus atributos.	0	
PROGRAMAÇÃO ORIENTADA A OBJETOS	Sprint 8 - Dashboard e e-mail	ONÇA PINTADA	DAIANA MACHADO DA COSTA	Boa noite! Estou com dificuldade para definir o que é uma classe externa, tipo explicar. Alguém pode me ajudar?	0	
PROGRAMAÇÃO ORIENTADA A OBJETOS	Sprint 8 - Dashboard e e-mail	ONÇA PINTADA	CRISTIANO SANGALLI FIEL	Boa noite Colegas! Aproveitando a dúvida do Arthur... está certa a minha resposta? PERGUNTA: Por que os atributos necessitam de métodos getters e setters? todos os atributos deverão ter getters e setters? quando sim, quando não? há getters ou setters não vinculados à atributos? RESPOSTA: Os métodos getters são responsáveis por retornar o valor dos atributos, já os métodos setters, são responsáveis por modificar o valor dos atributos. Porém, nem todos os atributos necessitam ter getters e setters. Como exemplo, cito a classe trabalhada em aula, chamada Televisao. Nesta classe o atributo volume vai aumentando o seu valor de acordo com o pressionar do botão de aumentar ou diminuir volume. Neste caso, o método setter de volume não deve ser criado, pois o atributo volume deve ir incrementando ou decrementando de atributo em atributo, e não receber um valor (atributo) fixo diretamente. Utilizar getters e setters ajuda a proteger os atributos, e também dá a possibilidade de fazer o encapsulamento dos dados.	4	
PROGRAMAÇÃO ORIENTADA A OBJETOS	Sprint 8 - Dashboard e e-mail	ONÇA PINTADA	ARTHUR CORBELLINI	Boa noite colegas, queria saber se está correta a minha resposta ou se há algo a acrescentar. Segue abaixo pergunta; seguida de minha resposta - PERGUNTA: por que os atributos de uma classe devem ser encapsulados, ou seja, por que devemos aplicar o modificador de acesso private nos atributos? - RESPOSTA: Pois utilizando o modificador "private", a integridade da classe é mantida. Ou seja, os atributos são fixos e só serão alterados e acessados quando a classe é criada ("newizada"), ou quando é feito um "set atributo".	2	

Fonte: da autora (2020)

Em relação ao planejamento e levantamento de requisitos, bem como o modelo de caso de uso, pode-se afirmar que todos os requisitos e casos de uso foram atendidos plenamente. O modelo do banco de dados planejado inicialmente sofreu algumas mudanças, bem como a criação de uma nova tabela para armazenamento dos dados que não estava planejada inicialmente. Também é importante destacar que os protótipos de interface confeccionados na fase de projeto foram de grande importância e facilitaram o desenvolvimento das telas do “EaseChat”.

O próximo capítulo tem o objetivo de detalhar a metodologia utilizada para a realização dos testes e os resultados obtidos.

## 5 RESULTADOS E DISCUSSÕES

Este capítulo detalha os resultados obtidos acerca da realização dos procedimentos citados no capítulo anterior. No processo de desenvolvimento algumas funcionalidades foram criadas buscando atingir o objetivo proposto neste trabalho. Dentre elas, a ferramenta de comunicação destinada aos alunos e professores, um algoritmo para comparar os dados coletados através da utilização do “EaseChat” e os textos já classificados anteriormente por profissionais. Neste algoritmo é feita a classificação das mensagens obtidas e a avaliação dos resultados. A utilização deste algoritmo de classificação de mensagens foi importante para auxiliar os professores a identificar os estudantes que possuem maiores dificuldades no aprendizado e realizar ações que contribuíssem para o seu desenvolvimento, tornando a ferramenta mais efetiva.

Para validar a solução proposta foi disponibilizado para os alunos de uma turma de Programação Orientada a Objetos e Projeto Integrador I da Universidade do Vale do Taquari - Univates a ferramenta de comunicação, colaboração e troca de mensagens entre os integrantes dos grupos de trabalho. A turma de Programação Orientada a Objetos é composta por 20 alunos e ficou distribuída nos grupos: “Onça pintada”, “Mamute dourado”, “Falcão negro” e “Águia azul”, ambas com 5 integrantes. Já a turma de Projeto Integrador I é composta por 21 alunos e ficou com a seguinte distribuição de grupos: “PI1 – 1” com 5 integrantes, “PI1 – 2” com 5 integrantes, “PI1 – 3” com 6 integrantes e “PI1 – 4” com 5 integrantes.

O Apêndice A foi disponibilizado para os alunos com todas as instruções para realizarem a instalação e utilização do *plug-in*. A ferramenta foi utilizada durante o período de vinte dias pelos grupos que foram organizados por critérios estabelecidos pelos professores das turmas, bem como os assuntos que foram abordados durante o uso da ferramenta. A solução foi utilizada no desenvolvimento de assuntos previstos na ementa das disciplinas.

O critério utilizado pelos professores para dividir as turmas em grupos foi estabelecido com base nas notas obtidas pelos alunos na primeira prova realizada na disciplina. Sendo assim, os professores distribuíram a turma em quatro grupos com alunos que tiraram nota alta, mediana e baixa. Isso foi importante para que houvesse equilíbrio de conhecimento e aptidão entre os membros dos grupos, facilitando a troca de conhecimentos entre eles.

As intervenções realizadas nas turmas foram feitas pelos professores que, durante o uso da ferramenta, frisaram a importância da interação entre os grupos e incentivaram o uso da ferramenta. Durante o período de utilização da ferramenta foi disponibilizado um *e-mail* de apoio para o envio de dúvidas e sugestões. Como o tempo para a utilização da ferramenta foi um período curto, não foram realizadas ações em relação aos alunos que não participaram ou realizaram poucas contribuições para as dúvidas dos colegas.

Com isso, buscou-se comparar o desempenho médio dos grupos da primeira turma com os grupos da segunda turma visando identificar diferenças de desempenho. O professor pode acessar em tempo real o sistema para analisar os dados trocados entre os membros dos grupos e pode fazer interferências caso julgue necessário. Após o período de utilização da solução proposta foi aplicado um questionário com os alunos da turma para constatar a percepção dos mesmos em relação à melhora, ou não, de desempenho do grupo. O Apêndice B contém o questionário aplicado com os alunos.

## **5.1 Coleta dos dados**

O primeiro passo para a realização do processo de mineração de textos foi a coleta de mensagens obtidas em um ambiente virtual de estudo, de turmas de semestres anteriores, da Universidade do Vale do Taquari. Esta etapa fez-se necessária para que as mensagens enviadas pudessem ser analisadas e classificadas por profissionais da área da Psicologia, conforme a presença ou não de um comportamento solícito. Após a análise e classificação das mensagens foi possível consolidar uma base de testes e treinamentos para ser utilizada no algoritmo de mineração de textos.

## 5.2 Pré-processamento e classificação

Nesta etapa foi aplicado o pré-processamento de dados, apenas nas mensagens que foram selecionadas para fazer parte deste estudo de caso. Foram removidos os caracteres especiais, conhecidos como *stopwords*<sup>1</sup>, prefixos e sufixos de palavras, com a finalidade de deixar o texto mais limpo para a realização da análise.

Também foram aplicadas técnicas de processamento de linguagem natural, a fim de tornar o texto mais limpo para as próximas etapas, através da utilização da biblioteca NLTK do Python. Foi criado um algoritmo que realiza todos os requisitos de pré-processamento, utilizando o processamento de linguagem natural, remoção de palavras que não contribuem para a classificação das frases coletadas a partir da base de treinamento utilizada. Para este trabalho, a implementação aceita como entrada para a base de treinamento um arquivo do tipo CSV com as frases de entrada.

No pré-processamento é realizada a decomposição das frases, ou seja, sentenças que contém um conjunto de letras que formam as palavras, números e símbolos que são agrupados em frases. Segundo Barbosa (2013), o ato de pré-processar esses elementos significa percorrer as frases mineradas a cada espaço contido tendo como limite a quebra de linha.

O comportamento analisado foi classificado como solícito ou não solícito. Como primeiro passo aplicou-se a limpeza do texto, que consistiu em aplicar a radicalização, remoção das *stopwords* e caracteres indesejados. Sendo assim, foi necessário fazer o *download* de *stopwords* do idioma *portuguese* da biblioteca NLTK e, na sequência, foi criada uma função que recebe o texto e extrai apenas as palavras que não estão contidas na lista de *stopwords* (*list\_stopwords*), insere as palavras no *array* de frases (*phrases*) já tratadas na *list\_stopwords* e retorna este *array*, conforme detalhado na Figura 22.

---

<sup>1</sup> Stopwords: são palavras irrelevantes para o entendimento do sentido de um texto, como por exemplo: as, e, os, para, entre outras. Essas palavras são, geralmente, removidas dos textos na etapa de pré-processamento dos dados

Figura 22 – Processo de remoção de *stopwords*

```
nltk.download('stopwords')
list_stopwords = nltk.corpus.stopwords.words('portuguese')
np.transpose(list_stopwords)

def removeStopwords(text):
    phrases = []
    for(words, feeling, student) in text:
        noStop = [p for p in words.split() if p not in list_stopwords]
        phrases.append(noStop, feeling, student)
    return phrases
```

Fonte: da autora (2020)

Em seguida, conforme Figura 23, aplicou-se a técnica *Stemming*, que consiste em remover prefixos e sufixos de palavras. Para Lustosa (2019) o *Stemming* é muito utilizado em mecanismos de buscas para indexação de palavras. Com esta técnica, ao invés de armazenar todas as formas de uma determinada palavra, um mecanismo de busca armazena apenas o *stem* da palavra, fazendo com que o tamanho de índice reduza e aumente a performance do processo de busca. A partir do pré-processamento, criou-se o modelo de classificação de solicitude e classificou-se todas as mensagens coletadas, analisadas, separadas e pré-processadas.

Para utilizar o *Stemming*, foi criada a função `aply_Stemmer` que recebe como parâmetro a mensagem a ser analisada e percorre todas as palavras aplicando o *Stemming* e armazenando essas palavras em um *array* (`phrases_without_Stemming`) que é retornado ao final.

Figura 23 – Aplicação da técnica *Stemming*

```
def aply_Stemmer(text):
    stemmer = nltk.stem.RSLPStemmer()
    phrases_without_Stemming = []
    for(words, sentimento, student) in text:
        with_Stemming = [str(stemmer.stem(p)) for p in words.split() if p not in list_stopwords]
        phrases_without_Stemming.append((with_Stemming, sentimento, student))
    return phrases_without_Stemming

phrases_with_Stem_training = aply_Stemmer(training_base)

pd.DataFrame(phrases_with_Stem_training, columns=['Frase', 'Sentimento', 'Aluno']).sample(50);
```

Fonte: da autora (2020)

### 5.3 Análise dos dados

Após o pré-processamento e classificação das mensagens compartilhadas entre os alunos, fez-se necessária a análise das classificações geradas. Através de um algoritmo de avaliação, baseado em frases classificadas por três profissionais da área da Psicologia, a classificação foi justificada. A frequência com que algumas palavras apareciam nas mensagens coletadas foi primordial para a demonstração dos resultados. Ao avaliar as 116 mensagens da base de treinamento, observou-se, conforme Figura 24, que foram classificadas 63 como solícitas (1) e 53 como não solícitas (0).

Figura 24 – Total de mensagens classificadas como solícitas e não solícitas

```
print('Tamanho da base de treinamento {}'.format(example.shape[0]))
example.Sentimento.value_counts()

Tamanho da base de treinamento 116
1      63
0      53
Name: Sentimento, dtype: int64
```

Fonte: da autora (2020)

Na Figura 25 observa-se que o algoritmo obteve uma precisão de 85,12%, totalizando 99 classificadas corretamente e 17 erroneamente. Esta precisão é chamada de acurácia e, levando em consideração que o tamanho da base de treinamento, é relativamente pequeno este resultado é muito satisfatório.

Figura 25 – Acurácia do algoritmo

```
print(nltk.classify.accuracy(classifier, base_completa_treinamento))

0.8512396694214877
```

Fonte: da autora (2020)





## 5.4 Análise dos dados da turma de Programação Orientada a Objetos


Nos quatro grupos em que a turma foi dividida é possível notar algumas semelhanças de comportamento. Entre elas, está o fato de que os alunos realizaram poucas perguntas e, consequentemente, contribuições.


Na Figura 26 é possível observar a análise de solicitude do grupo “Águia Azul”. O grupo realizou 4 perguntas durante o uso da ferramenta, entretanto apenas 2 respostas foram enviadas. Para o algoritmo minerador uma das perguntas realizadas do no grupo é considerada solícita e nenhuma das duas respostas realizadas são de caráter solícito. Com base nesses números pode-se perceber que este grupo não possui uma boa socialização e as contribuições realizadas no “EaseChat” não são de relevância para o aprendizado dos colegas.

Figura 26 – Análise de solicitude do grupo Águia Azul

TOTAL DE MENSAGENS  
6

TOTAL DE PERGUNTAS  
4

TOTAL DE RESPOSTAS  
2

ALUNOS  
5

ANÁLISE DE SOLICITUDE

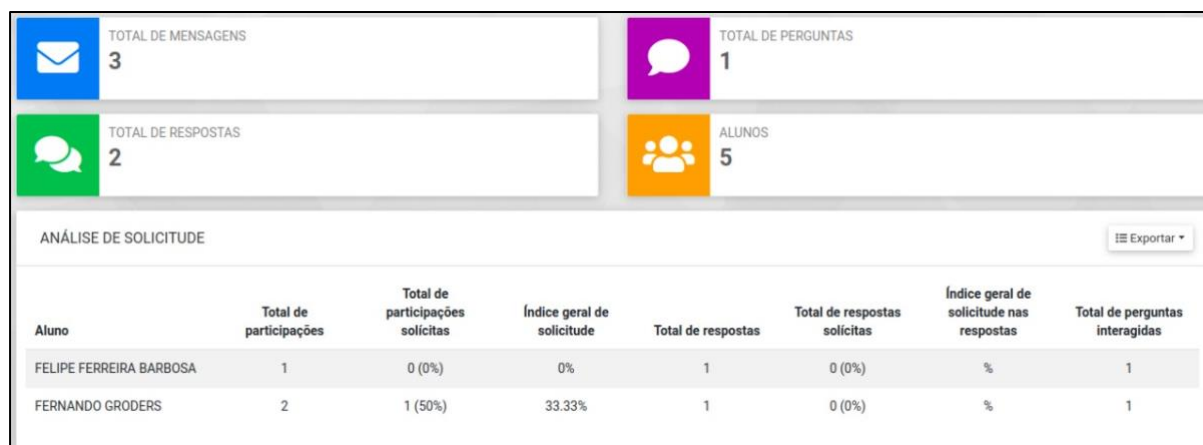
Exportar

Aluno	Total de participações	Total de participações solícitas	Índice geral de solicitude	Total de respostas	Total de respostas solícitas	Índice geral de solicitude nas respostas	Total de perguntas interagidas
CARLO LORENZONI MARX	1	0 (0%)	0%	1	0 (0%)	%	1
FABRÍCIO GERHARDT JÚNIOR	1	1 (100%)	16.67%	0	0 (0%)	%	0
JOÃO PEDRO MULLER LIMA	2	0 (0%)	0%	1	0 (0%)	%	1
LUCAS MATHEUS OLIVEIRA DA SILVA	1	0 (0%)	0%	0	0 (0%)	%	0
LUCCA KEUNECKE ISSE	1	0 (0%)	0%	0	0 (0%)	%	0

Fonte: da autora (2020)

Na Figura 27 é apresentada a análise do grupo “Falcão Negro”, onde foi realizada 1 pergunta que obteve 2 respostas. O aluno que realizou 2 participações teve uma delas considerada solícita, entretanto as respostas enviadas não foram consideradas solícitas. Desta forma, assim como no grupo anterior, as contribuições realizadas no grupo não possuem relevância em relação ao que foi questionado no grupo.

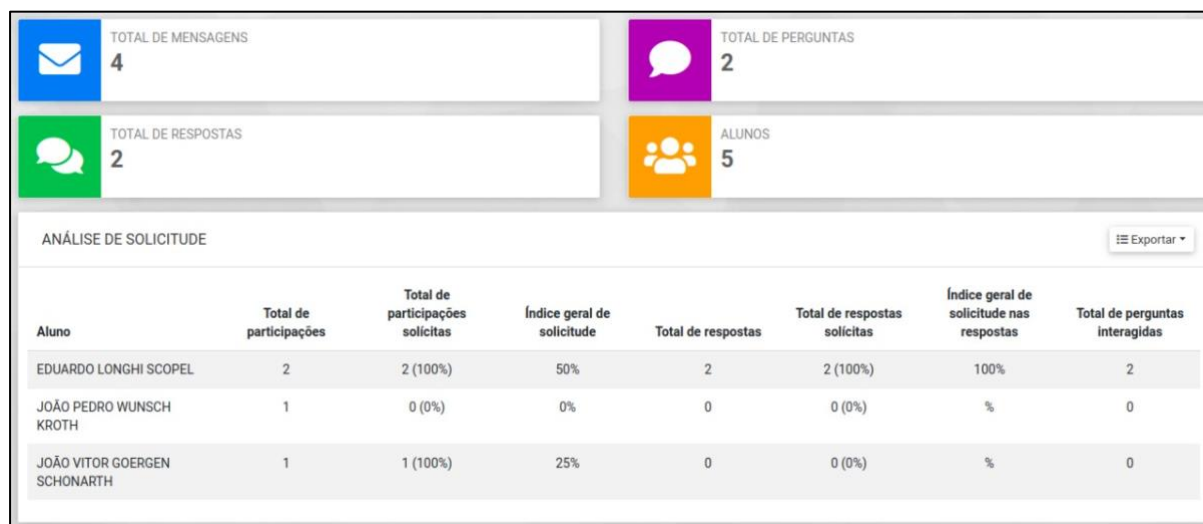
Figura 27 – Análise de solicitude do grupo Falcão Negro



Fonte: da autora (2020)

Os dados do grupo “Mamute Dourado” são apresentados na Figura 28. O grupo realizou 2 perguntas e 2 respostas. Neste grupo as duas respostas foram consideradas solícitas e foram feitas pelo mesmo aluno, indicando que este é um aluno participativo e pode contribuir para o aprendizado dos colegas.

Figura 28 – Análise de solicitude do grupo “Mamute Dourado”




Fonte: da autora (2020)


O último grupo que foi analisado foi o grupo “Onça Pintada”, conforme Figura 29. Este grupo realizou 9 perguntas e 14 respostas e, em comparação com os outros grupos, pode-se perceber que é um grupo que realiza mais interações. Destas interações, percebe-se que todos

os alunos realizaram participações que, em sua maioria, são solícitas levando em considerações as perguntas realizadas. Ou seja, embora os alunos façam perguntas estas também contêm informações e posicionamentos solícitos. Entretanto, mesmo tendo um número de interações maior que o dos outros grupos, a solicitude nas respostas não foi muito significativa. Das 14 respostas realizadas no grupo, apenas 4 foram consideradas solícitas.


Figura 29 – Análise de solicitude do grupo Onça Pintada




TOTAL DE MENSAGENS  
23



TOTAL DE PERGUNTAS  
9



TOTAL DE RESPOSTAS  
14



ALUNOS  
5

ANÁLISE DE SOLICITUDE

Exportar

Aluno	Total de participações	Total de participações solícitas	Índice geral de solicitude	Total de respostas	Total de respostas solícitas	Índice geral de solicitude nas respostas	Total de perguntas interagidas
ARTHUR CORBELLINI	7	3 (42.86%)	13.04%	3	2 (66.67%)	14.29%	3
CRISTIANO SANGALLI FIEL	5	0 (0%)	0%	4	0 (0%)	%	4
DAIANA MACHADO DA COSTA	2	1 (50%)	4.35%	1	0 (0%)	%	1
ENZO WEINBERGER MOSMAN	5	1 (20%)	4.35%	3	0 (0%)	%	3
RAFAEL MICHEL DOCENA	4	3 (75%)	13.04%	3	2 (66.67%)	14.29%	3

Fonte: da autora (2020)

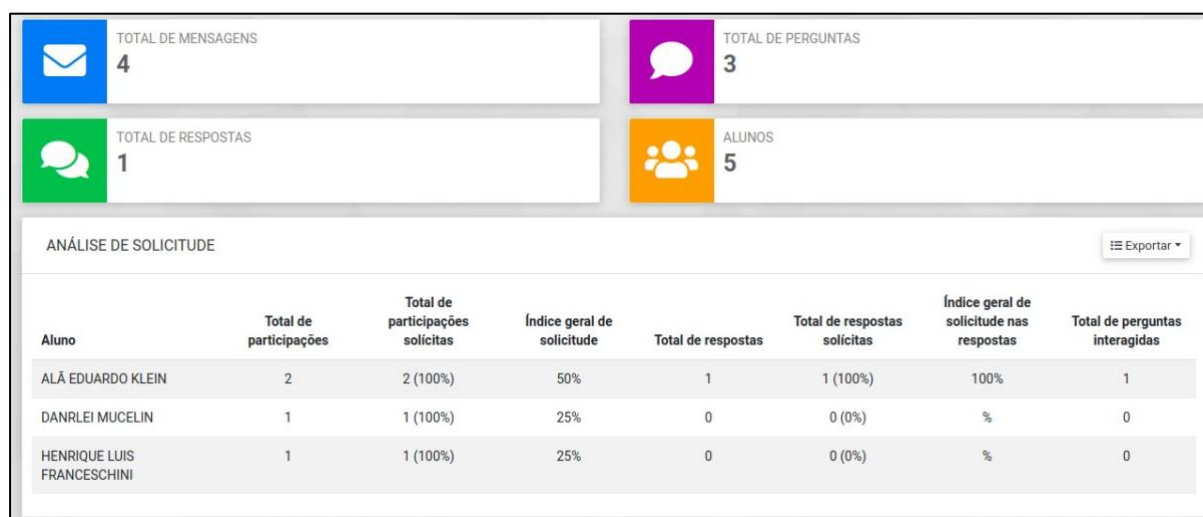
O fato de os grupos não terem realizado muitas interações dificultou, de certa forma, a análise dos dados e um dos motivos para a pouca interação pode estar relacionado com o pouco tempo de uso da ferramenta. Outro fator a ser considerado diz respeito aos grupos possuírem poucos integrantes, tendo em vista que em grupos maiores o número de participações possivelmente cresceria. Entretanto, já é possível ver que a postura de alguns alunos é de contribuir com as dúvidas e com o aprendizado dos colegas.

## 5.5 Análise dos dados da turma de Projeto Integrador I

A Figura 30 apresenta os dados do grupo “PII – 1”, que realizou 3 perguntas e 1 resposta. Este grupo possui pouca interação, entretanto toda interação de pergunta e resposta

realizada foi considerada solícita. Este dado indica que o grupo possui um bom nível de participação e, sendo estimulado, pode realizar mais interações relevantes para o aprendizado dos colegas de grupo.

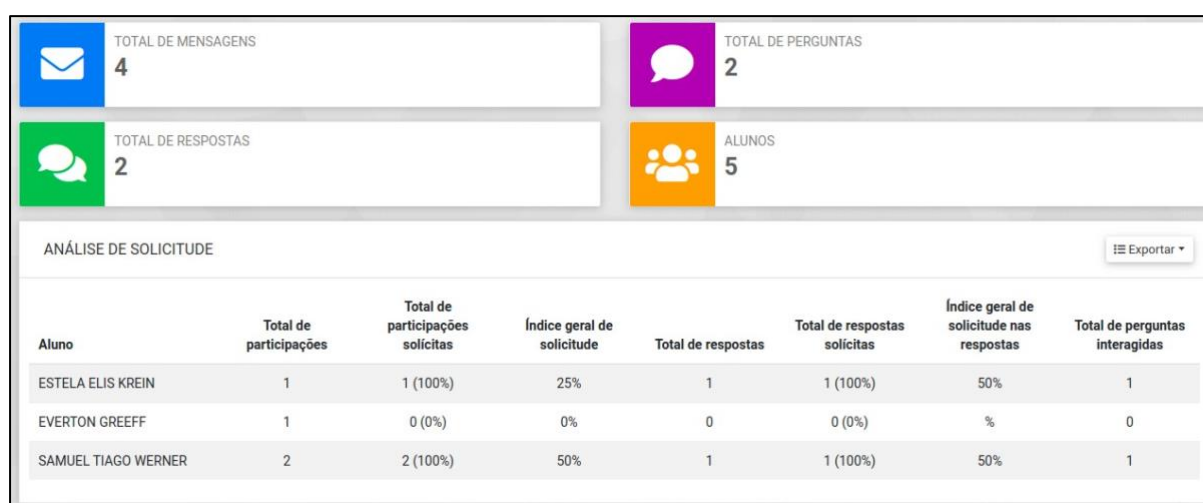
Figura 30 – Análise de solicitude do grupo PI1 – 1



Fonte: da autora (2020)

Os dados do grupo “PI1 – 2” são apresentados na Figura 31. O grupo realizou 4 participações sendo 2 perguntas e 2 respostas. Todas as participações do grupo foram consideradas solícitas e mostra que o grupo é totalmente participativo.

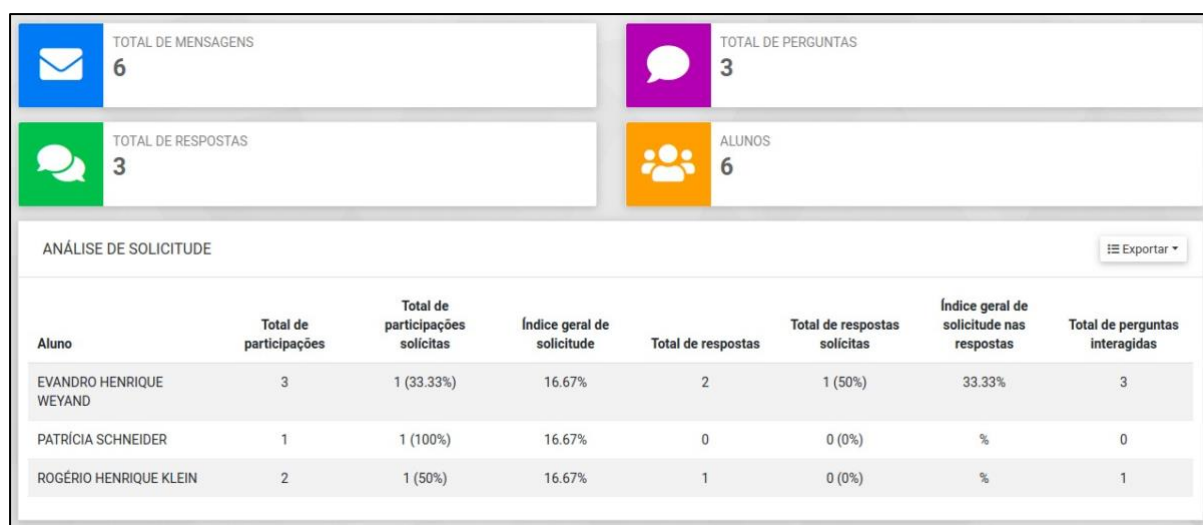
Figura 31 – Análise de solicitude do grupo PI1 – 2



Fonte: da autora (2020)

A Figura 32 apresenta os dados do grupo “PI1 – 3”. Este grupo realizou 6 participações, sendo elas 3 perguntas e 3 respostas. Em comparação com o grupo anterior, este grupo não foi tão solícito em suas respostas, tendo apenas 1 resposta solícita. Mesmo que o grupo não tenha realizado tantas respostas solícitas, demonstra que os membros estão engajados em ajudar os colegas.

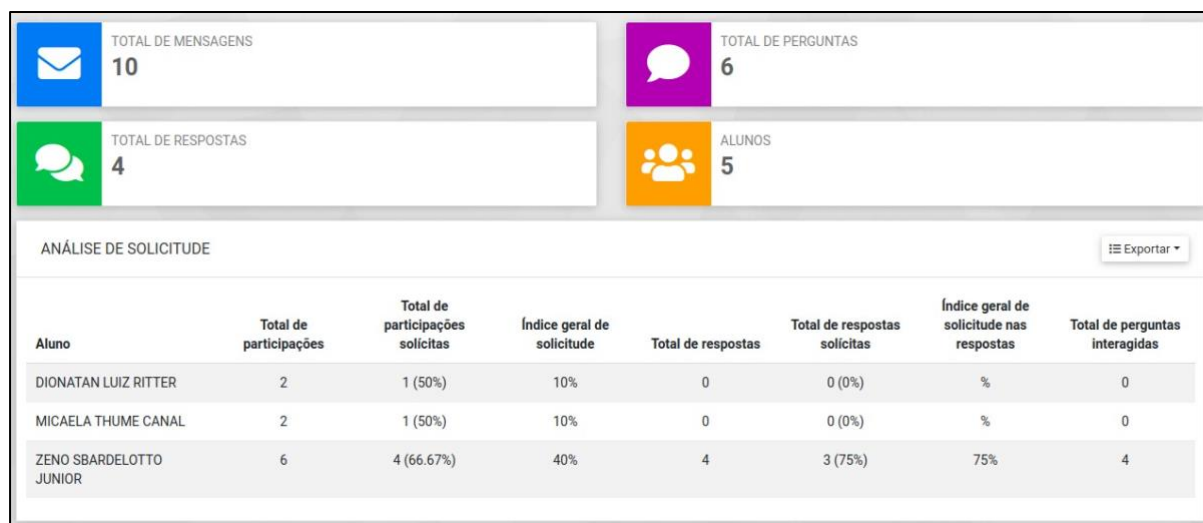
Figura 32 – Análise de solicitude do grupo PI1 – 3



Fonte: da autora (2020)

O último grupo que foi analisado foi o grupo “PI1 – 4”, que apresentou 10 interações sendo elas 6 perguntas e 4 respostas. Os dados podem ser vistos na Figura 33. Neste grupo percebe-se que apenas um aluno contribuiu com respostas para as dúvidas dos colegas. Das 4 contribuições 3 delas foram consideradas solícitas, indicando que este aluno é bastante participativo e suas contribuições, de fato, podem auxiliar os colegas.

Figura 33 – Análise de solicitude do grupo PI1 – 4



Fonte: da autora (2020)

Os grupos da turma de Projeto Integrador I, em comparação com os grupos da turma de Programação Orientada a Objetos também possuem um nível baixo de interações, possivelmente pelo número pequeno de participantes em cada grupo. Entretanto, mesmo com um número pequeno de interações elas são, em sua maioria, classificadas como solícitas, ou seja, que sejam efetivas para o aprendizado dos colegas.

Desta forma, acredita-se que, com o uso da ferramenta a longo prazo, a quantidade de interações, possivelmente, seria maior e a tendência é que o nível de solicitude também aumente. Embora tenham sido poucos dados para analisar, o resultado foi satisfatório e significativo para o alcance dos objetivos deste trabalho. A baixa interação entre os alunos já era prevista, levando em consideração o tempo que foi disponibilizado para a utilização da ferramenta, porém, não houve prejuízo na análise realizada.

## 5.6 Análise do questionário aplicado

Para poder avaliar a ferramenta proposta foi elaborado um questionário com seis questões subjetivas e objetivas, exibidas no Apêndice B deste trabalho. O questionário foi disponibilizado para os 41 alunos das duas turmas analisadas e obteve o total de 18 respostas.

A questão um tinha o objetivo de captar a percepção dos alunos quanto à importância de uma ferramenta colaborativa para o aprendizado. Conforme o esperado, as respostas foram bastante positivas, onde, em uma escala de 1 a 5, para 6 alunos a importância é de nível 4 e para 10 alunos a importância é de 5, como pode ser visto na Figura 34.

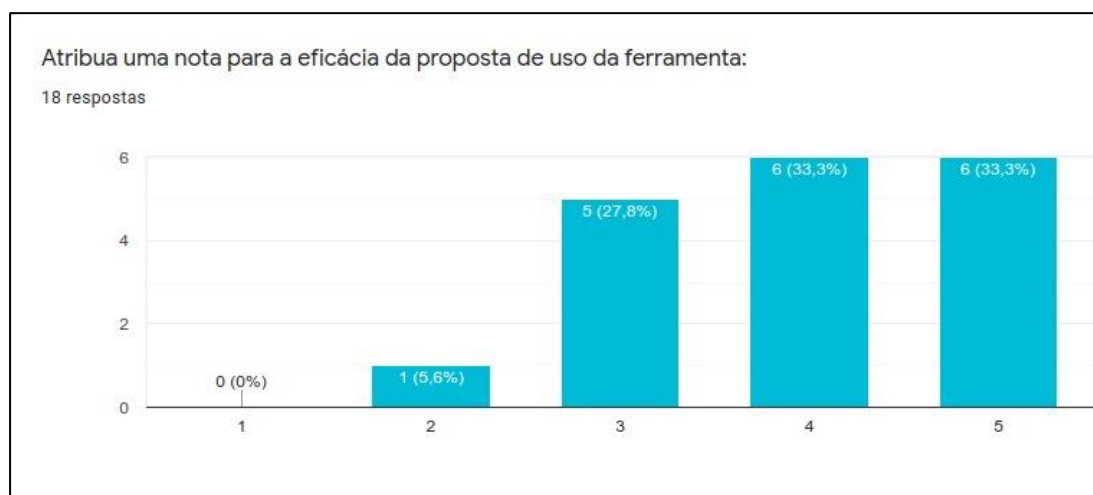
Figura 34 – Questão um do questionário, referente a importância de ferramentas colaborativas



Fonte: da autora (2020)

A Figura 35 apresenta a questão dois que buscava avaliar a eficácia da ferramenta proposta, ou seja, se esta atendia o objetivo proposto de ser um meio de comunicação entre os estudantes. O resultado foi positivo, totalizando 70,6% das respostas nos níveis 4 e 5, que consideram alta a eficácia da ferramenta.

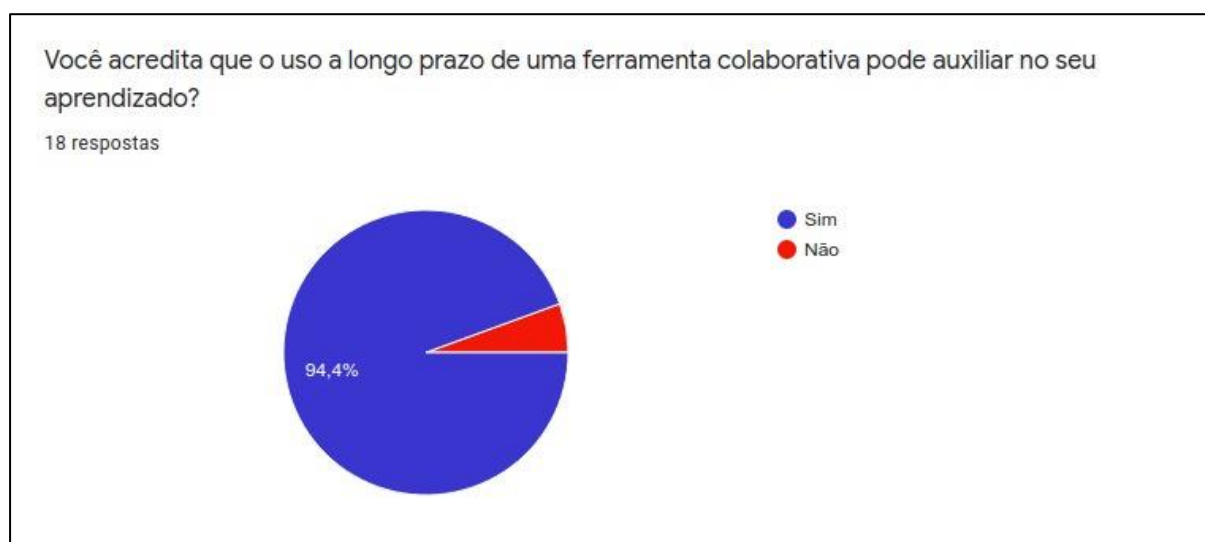
Figura 35 – Questão dois, referente a eficácia da ferramenta



Fonte: da autora (2020)

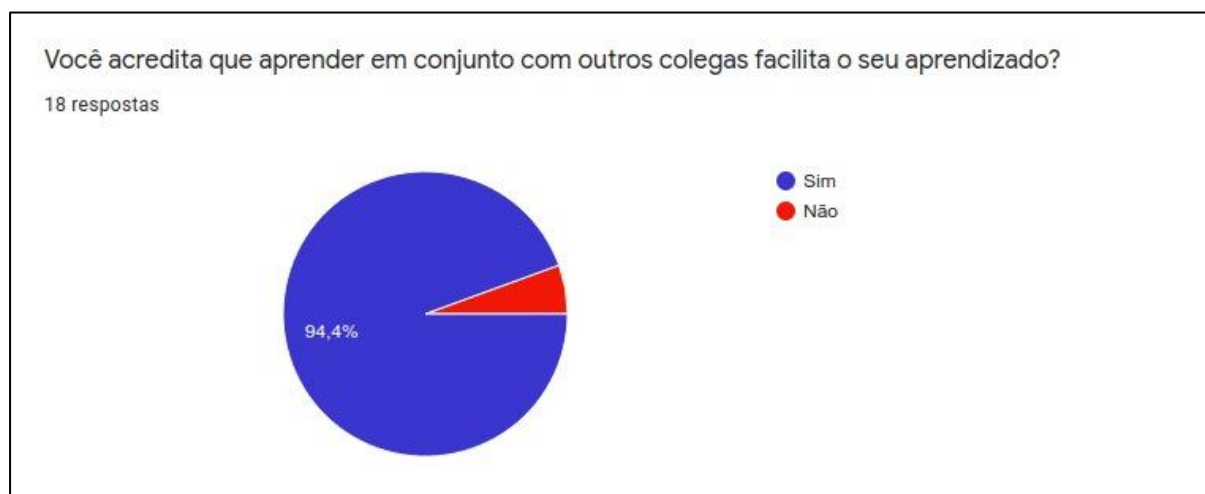
Nas questões três e quatro objetivou-se identificar se a utilização de uma ferramenta de colaboração a longo prazo pode contribuir no aprendizado dos estudantes e se o fato de aprender em conjunto com outros colegas é um fator importante. Para 94,1% dos alunos que responderam o questionário as respostas foram positivas, conforme é possível observar nas Figuras 36 e 37.

Figura 36 – Questão três, referente ao uso a longo prazo da ferramenta colaborativa



Fonte: da autora (2020)

Figura 37 – Questão quatro, referente ao aprendizado em conjunto



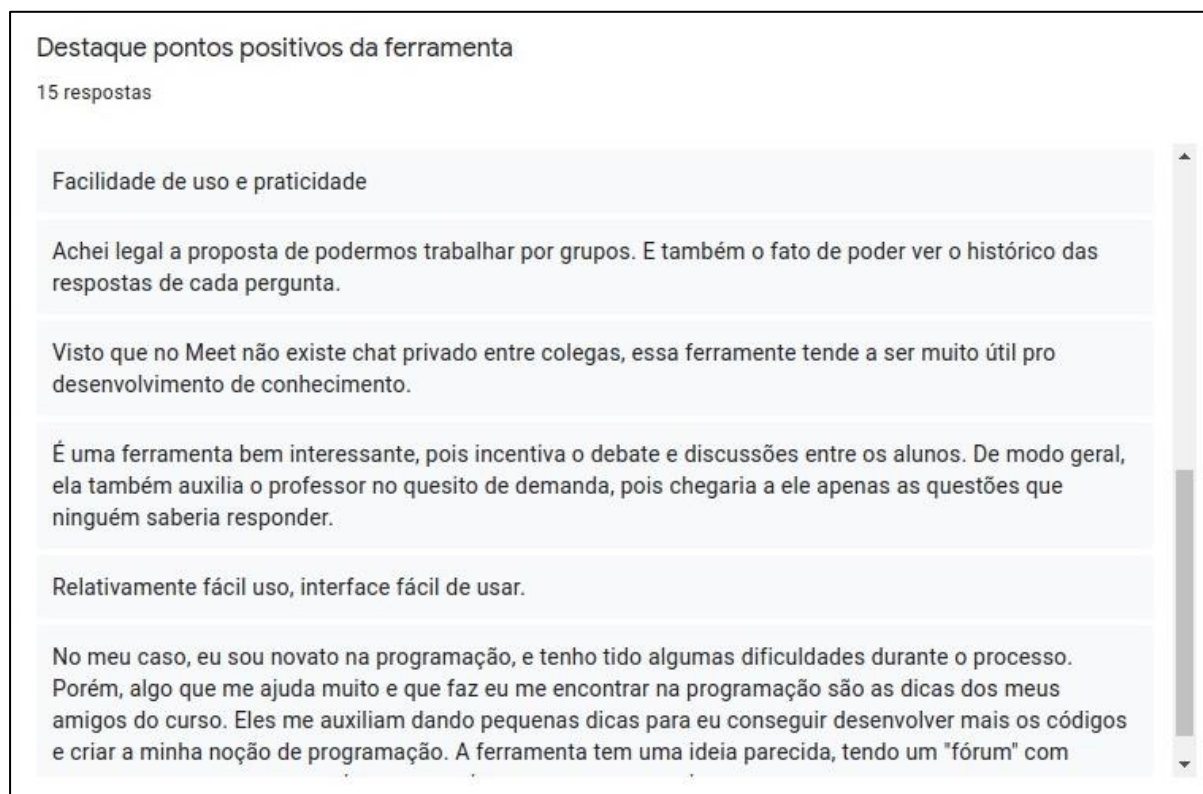
Fonte: da autora (2020)

Para complementar as questões e finalizar o questionário, as questões cinco e seis solicitaram aos estudantes a explanação de pontos positivos e negativos da ferramenta



“EaseChat”. Estas questões permitem capturar a opinião dos alunos, aviso de erros e sugestões de novas funcionalidades que agreguem a ferramenta proposta. As respostas dos alunos podem ser vistas nas Figuras 38 e 39.

Figura 38 – Questão cinco, referente aos pontos positivos da ferramenta



**Destaque pontos positivos da ferramenta**  
15 respostas

Facilidade de uso e praticidade

Achei legal a proposta de podermos trabalhar por grupos. E também o fato de poder ver o histórico das respostas de cada pergunta.

Visto que no Meet não existe chat privado entre colegas, essa ferramenta tende a ser muito útil pro desenvolvimento de conhecimento.

É uma ferramenta bem interessante, pois incentiva o debate e discussões entre os alunos. De modo geral, ela também auxilia o professor no quesito de demanda, pois chegaria a ele apenas as questões que ninguém saberia responder.

Relativamente fácil uso, interface fácil de usar.

No meu caso, eu sou novato na programação, e tenho tido algumas dificuldades durante o processo. Porém, algo que me ajuda muito e que faz eu me encontrar na programação são as dicas dos meus amigos do curso. Eles me auxiliam dando pequenas dicas para eu conseguir desenvolver mais os códigos e criar a minha noção de programação. A ferramenta tem uma ideia parecida, tendo um "fórum" com

Fonte: da autora (2020)

Como pontos positivos da ferramenta foi bastante destacada a facilidade e a praticidade para utilizar o “EaseChat”, bem como a proposta da ferramenta de proporcionar o trabalho em grupos e a possibilidade de visualizar o histórico de respostas de cada pergunta. Ainda nesta questão foi ressaltada a importância das dicas e sugestões dos colegas no processo de aprendizado.

Figura 39 – Questão seis, referente aos pontos negativos da ferramenta

**Destaque pontos negativos da ferramenta**

13 respostas

NA.

Por estar em seus estágios iniciais, acredito, achei meio difícil de se "encontrar" nas perguntas e respostas.

Não encontrei

Na minha opinião, para a ferramenta ficar 100% chamativa, seria necessário tirar este sistema de grupos e deixar toda a turma para se comunicar, não sei a logística por trás dos grupos, porém na minha opinião iria ajudar na velocidade da resposta e na quantidade. E também, a questão do anonimato, eu já mandei perguntas na ferramenta e não liguei muito para o meu nome escrito, porém acredito que tenham alunos tímidos com receio de mandar mensagens que façam seus colegas acharem engraçado, ou algo do tipo.

Só é compatível com uma versão mais antiga do netbeans

Não encontrei nenhum ponto negativo.

Não dá pra saber quem está no grupo que foi criado, necessita de uma forma de anexar imagens.

Fonte: da autora (2020)

Sobre os pontos negativos em relação a ferramenta, foi citado o fato de não ser possível visualizar os colegas de grupo, a incompatibilidade do *plugin-in* em IDEs mais novas e a utilização de grupos. Foi sugerido um único grupo para que todos os colegas pudessem ver as perguntas e respondê-las. Muitos alunos não destacaram pontos negativos na ferramenta, algo que é considerado satisfatório.

Após finalizado o período de aplicação do questionário, o resultado obtido com o uso da ferramenta confirma a hipótese de que um ambiente colaborativo pode contribuir para o aprendizado dos estudantes. Destaca-se a percepção dos alunos em relação a importância da ferramenta e o fato de que eles continuariam utilizando-a. Desta forma pode-se afirmar que a ferramenta “EaseChat” teve um desempenho positivo e uma boa aceitação dos estudantes.

## 6 CONCLUSÕES

Atualmente, ainda existem grandes carências no apoio ao aprendizado de lógica de programação. Esta dificuldade faz com que muitos estudantes desistam dos cursos de TI. Sendo assim, é importante buscar alternativas que estimulem e facilitem o aprendizado dos estudantes. Neste contexto, a utilização de ferramentas colaborativas para a troca de dúvidas e soluções pode resolver ou amenizar o problema de aprendizagem. Entretanto, mesmo utilizando este recurso, muitas vezes, as dúvidas compartilhadas nos fóruns de discussão e em ferramentas colaborativas não são respondidas ou são respondidas de maneira inadequada, persistindo as dificuldades.

Esses dilemas culminaram no objetivo deste trabalho, avaliar a eficiência de uma ferramenta colaborativa, integrada ao ambiente de desenvolvimento, na aprendizagem da programação de computadores. Através da pesquisa realizada, a importância e relevância da ferramenta de forma integrada, mostra-se significativa na aprendizagem e resolução de problemas relacionados à lógica de programação e desenvolvimento.

O processo de desenvolvimento da ferramenta foi simplificado devido ao consistente planejamento, uma vez que a definição dos requisitos e dos modelos de caso de uso foram bem detalhados. Entretanto, o processo de desenvolvimento do *plug-in* para ser utilizado na IDE foi complicado e limitado. Existem poucos materiais acerca do tema e as novas versões da IDE não comportam os componentes utilizados. Sendo assim, alguns alunos não conseguiram utilizar o “EaseChat” como um *plug-in*, somente a versão *web*.

Para o desenvolvimento do trabalho diversas pesquisas acerca do tema análise de sentimentos e solicitude foram realizadas e, com isso, é possível comprovar a escassez de pesquisas no que diz respeito à análise de solicitude. Sendo assim, foi fundamental a realização de uma coleta prévia de mensagens e a análise e classificação realizada por profissionais da

área da Psicologia, a fim de elaborar uma base de testes confiável para a realização dos testes.

Com a base de dados para testes montada e utilizada pôde-se perceber que havendo uma base de treinamento maior, classificada por profissionais da Psicologia, consegue-se treinar uma base melhor estruturada e capaz de avaliar com maior precisão os dados obtidos com a troca de mensagens. Finalmente, considerando as técnicas de mineração de textos pode-se melhorar em alguns aspectos como, por exemplo, a lista de *Stopwords* na língua portuguesa da biblioteca NLTK, que não contém todas as palavras necessárias para a realização do pré-processamento dos dados.

Conclui-se que, por meio dos mecanismos utilizados para validar a hipótese deste trabalho, que uma ferramenta colaborativa pode auxiliar no aprendizado dos estudantes. Entretanto, ainda podem ser realizadas melhorias na ferramenta “EaseChat” que melhorem o uso dos estudantes. É possível elencar melhorias na ferramenta, como o aumento da base de dados para treinamento do algoritmo minerador, com a finalidade de melhorar a assertividade e acurácia do mesmo, utilizar o “EaseChat” a longo prazo com mais turmas de programação. Uma possibilidade seria instituir seu uso nas disciplinas dos cursos de TI da Univates para aumentar a familiaridade dos estudantes com a ferramenta e aumentar as possibilidades de análise dos professores. Abranger o uso do “EaseChat” de forma que outros algoritmos de classificação possam ser considerados para utilização, melhorar as metodologias utilizadas para os testes levando em consideração as dificuldades apresentadas neste trabalho, aumentar a compatibilidade do *plug-in* com IDEs mais novas.

Através deste trabalho novos estudos podem ser realizados acerca a análise de solicitude, assim, contribuindo para a melhoria e aperfeiçoamento de ferramentas neste segmento. Desta forma, a utilização de ferramentas colaborativas tende a aumentar e, de fato, contribuir para o aprendizado dos estudantes. Espera-se, também, que estas ferramentas melhorem a percepção dos mesmos sobre os cursos de TI, diminuindo a evasão dos cursos e contribuindo para a formação de profissionais cada vez mais capacitados.

## REFERÊNCIAS

ASCENCIO, Ana F. G.; CAMPOS, Edilene A. V. de. **Fundamentos da programação de computadores: algoritmos, PASCAL, C/C++ (padrão ANSI) e JAVA**. 3. ed. São Paulo, SP: Pearson, 2012.

AUDY, Jorge L. N.; ANDRADE, Gilberto K. de; CIDRAL, Alexandre. **Fundamentos de sistemas de informação**. Porto Alegre: Bookman, 2007.

AZEVEDO, Breno F. T.; BEHAR, Patricia A.; REATAGUI, Eliseo B. **Análise das mensagens de fóruns de discussão através de um software para mineração de textos**. Aracajú, 2011.

AZEVEDO, Breno F. T.; BEHAR, Patricia A.; REATAGUI, Eliseo B. **Aplicação da mineração de textos em fóruns de discussão**. Porto Alegre, RS, 2010.

BIRD, Steven; KLEIN, Ewan; LOPER, Edward. **Natural Language Processing with Python – Analyzing Text with the Natural Language Toolkit**. 1 ed. 2019

BONATTI, Denilson. **Desenvolvimento de Sites Dinâmicos com Dreamweaver CC**

BRAGA, Luiz O. P. **Mídias são parciais! Será? Uma ferramenta de análise automática dos canais de notícia em relação às figuras políticas no Brasil**. Universidade Federal Rural de Pernambuco, 2017.

BRITO, Edivaldo. **Java: Entenda para que serve o software e os problemas da sua ausência**, 2014. Disponível em: <<https://www.techtudo.com.br/dicas-e-tutoriais/noticia/2014/11/java-entenda-para-que-serve-o-software-e-os-problemas-da-sua-ausencia.html>> Acesso em: 24 mai. 2020.

CARVALHO, Saullo; BRANCO, Castelo. **KExploit – Script em Python para automação de exploração de Kernel de Linux**. Brasília, DF, 2016

CECI, Flávio. **Um modelo semiautomático para a construção e manutenção de ontologias a partir de bases e documentos não estruturados**. 2010

CHAGAS, Lucinéia B. da C.; OLIVEIRA, Márcia G. de; TAVARES, Orivaldo de L. **APP – Uma arquitetura pedagógica para aprendizagem de programação de computadores**. Vitória, ES, 2016.

CHIAVENATO, Idalberto. **Iniciação a sistemas, organização e métodos SO&M**. Barueri, SP: Manole, 2010.

DALL'OGGIO, Pablo. **PHP: programando com orientação a objetos**. 3. ed. São Paulo: Novatec, 2016.

DALL'OGGIO, Pablo. **Adianti Framework para PHP**. 5. ed. São Paulo: Novatec, 2015.

DRINGUS, L. P.; ELLIS, T. **Using data mining as a strategy for assessing asynchronous discussion forums**. Computers & Education, 2005.

DUARTE, Felipe R. B. **Integração de Sistemas**, 2009. Disponível em: <<https://repositorio.ufsc.br/handle/123456789/184496>> Acesso em: 06 jun. 2020

FELDMAN, Ronen; SANGER, James. **The text mining handbook: advanced approaches in analyzing unstructured data**. Cambridge University Press, 2007.

FLATSCHART, Fabio. **HTML 5 - Embarque Imediato**. 1. ed. Rio de Janeiro, RJ: Brasport, 2011.

FORBELLONE, Andre L. V.; EBERSPACHER, Henri F. **Lógica de programação: a construção de algoritmos e estruturas de dados**. 3. ed. São Paulo, SP: Prentice Hall, 2013.

GAHAGAN, Judy; D'ALMEIDA, Eduardo. **Comportamento interpessoal e de grupo**. Rio de Janeiro: Zahar, 1976.

GIL, Antonio C. **Métodos e técnicas de pesquisa social**. 6. ed. São Paulo: Editora Atlas SA, 2008.

GOMES, Helder J. C. **Text Mining: análise de sentimentos na classificação de notícias**. Lisboa, 2019

GUEDES, Sérgio. **Lógica de programação algorítmica**. 1. E'd. São Paulo, SP: Pearson, 2014.

HENRI, France; BASQUE, Josianne. **Conception d'activités d'apprentissage collaboratif en mode virtuel**. In Deaudelin, Colette, & Nault, Thérèse. Sainte-Foy, Canada: Presses de l'Université du Québec, coll, 2003.

JOHNSON, David. W.; JOHNSON, Roger. T.; STANNE, Mary. B.; GARIBALDI, A. **Impact of group processing on achievement in cooperative groups**. J Soc Psycho, 1990.

JORGE, Marcos. **Java: passo a passo Lite**. 1. ed. São Paulo, SP: Pearson, 2004.

KRAEMER, Fábio L. **Comunicação, interação e aprendizagem: o fórum de discussão como estratégia de ensino**. Lajeado, RS, 2015.

KOTHAPALLI, M; SHARIFAHMADIAN, E; SHIH, L. **Data mining of Social Media for Analysis of product review**. 2016

KUEHNE, Bruno T. **Modelos e algoritmos para composição de Web Services com**

**qualidade de serviço**, 2009. Disponível em:

<<https://teses.usp.br/teses/disponiveis/55/55134/tde-05042010-111224/publico/bruno.pdf>>

Acesso em: 06 jun. 2020

LAAL, Marjan; LAAL, Mozghan. **Collaborative learning**: what is it?. Procedia – Social and Behavioral Sciences, 2011.

LAHIRI, Rochana. Comparison of Data Mining and Statistical Techniques form classification model. Thesis of Master of Science at the Louisiana State University, US. 2016. Disponível em: <[http://etd.lsu.edu/docs/available/etd-11012006-192748/unrestricted/Lahiri\\_thesis.pdf](http://etd.lsu.edu/docs/available/etd-11012006-192748/unrestricted/Lahiri_thesis.pdf)> Acesso em: 05 out. 2020

LAKATOS, Eva M.; MARCONI, Marina de A. **Fundamentos de metodologia científica**. 7. ed. São Paulo: Atlas, 2010.

LARA, Luiz G. **Administração, sistemas e ambientes**. Editora Intersaberes, 2019.

LUSTOSA, Alexandre G. **Análise de sentimentos em Python**, 2019. Disponível em:

<<https://mediumcom/@alegeorgelustosa/análise-de-sentimentos-em-python-2a7d04a836e0>>

Acesso em: 01 set. 2020

MACHADO, Leonardo D. P.; BERKENBROCK, Carla D. M.; ANSELMO, Guilherme; SIPLE, Ivanete Z. **Uma ferramenta colaborativa para apoiar a aprendizagem de programação de computadores**. Passo Fundo, RS: Editora UPF, 2018.

MARCELO, Antonio. **Apache**: configurando o servidor WEB para linux. São Paulo: Brasport, 2005.

MILANI, André. **PostgreSQL: guia do programador**. São Paulo: Novatec, 2008.

MORAIS, Edison A. M.; AMBRÓSIO, Ana P. L. **Mineração de textos**, 2007. Disponível em: <[http://ww2.inf.ufg.br/sites/default/files/uploads/relatorios-tecnicos/RT-INF\\_005-07.pdf](http://ww2.inf.ufg.br/sites/default/files/uploads/relatorios-tecnicos/RT-INF_005-07.pdf)> Acesso em: 25 mai. 2020

MORAN, José M.; MASETTO, Marcos T.; BEHRENS, Marilda A. **Novas tecnologias e mediação pedagógica**. Campinas, SP: Papirus, 2015.

MOSCOVICI, Fela. **Desenvolvimento interpessoal**: treinamento em grupo. 2. ed. Rio de Janeiro, RJ: Editora José Olympio, 2013.

MUNHOZ, Antonio S. **Objetos de aprendizagem**. Curitiba: Intersaberes, 2012.

NETBEANS. **Visão geral do NetBeans IDE**, 2020. Disponível em:

<[https://netbeans.org/features/index\\_pt\\_BR.html](https://netbeans.org/features/index_pt_BR.html)>. Acesso em: 24 mai. 2020.

NOSCHANG, Luis F.; PELZ, Fillipi; RAABE, A. **Portugolstudio**: Uma ide para iniciantes em programação. Anais do CSBC/WEI, 2014.

PRODANOV, Cleber A.; FREITAS, Ernani C. **Metodologia do Trabalho Científico: Métodos e Técnicas da Pesquisa e do Trabalho acadêmico**. 2. ed. Novo Hamburgo Rio Grande do Sul: Universidade FEEVALE, 2013.

PUGA, Sandra; RISSETTI, Gerson. **Lógica de programação e estruturas de dados**. 3. ed. São Paulo, SP: Pearson, 2016.

PYTHON. **General Python FAQ – Python 3.6 documentation**. Disponível em: <<https://docs.python.org/3/faq/general.html>> Acesso em: 02 set. 2020.

RANGEL, Mary. **Métodos de ensino para a aprendizagem e a dinamização das aulas**. 6.ed. Campinas, SP: Papirus, 2010.

REZENDE, Solange O.; MARCACINI, Ricardo M.; MOURA, Maria F. **Revista de Sistemas de Informação da FSMA**. 2011, p 7-21.

RIBEIRO, Marco A. de P. **Técnicas de aprender**. 1. ed. Petrópolis, RJ: Vozes, 2012.

SANTOS, Wilian P. S. **Análise dos Tweets sobre a Black Friday através da Mineração de Texto e Análise de sentimentos**. Rio de Janeiro, RJ: 2016.

SAKAMOTO, Cleusa K.; SILVEIRA, Isabel O. **Como fazer projetos de Iniciação Científica**. 1. ed. Pia Sociedade de São Paulo-Editora Paulus, 2019.

SELEME, Roberto B.; MUNHOZ, Antonio S. **Criando universidades corporativas no ambiente virtual**. São Paulo: Pearson Prentice Hall, 2011.

SCHENKER, Adam. **Graph-Theoretic Techniques for Web Content Mining**. University of South Florida, Florida, 2003.

SCHERER, Klaus R. **Psychological models of emotion**. 1. ed. 2005.

TAMAI, Ana L. M. **Modelos de predição – Naive Bayes**. Disponível em: <<https://medium.com/turing-talks/turing-talks-16-modelo-de-predicao-naive-bayes-6a3e744e7986>> Acesso em: 01 out. 2020.

TORRES, Patrícia L.; IRLA, Esrom A. F. **Aprendizagem colaborativa: teoria e prática**. 1. ed. Curitiba: Senar, 2014.

TURBAN, Efraim; et al. **Tecnologia da Informação para Gestão: transformando os negócios na economia digital**. 6 ed. Bookman, 2010.

VALENTINI, Carla B.; SOARES, Eliana M. do S. **Aprendizagem em ambientes virtuais: compartilhando ideias e construindo cenários**. Caxias do Sul, RS: Educs, 2010.

VAN DER WALT, Stéfani, COLBERT, S. Chris; VAROQUAUX, Gael. **The NumPy Array: A Structure for Efficient Numerical Computation**. Computing in Science & Engineering. v. 3. 2011. Disponível em: <<https://ieeexplore.ieee.org/document/5725236/authors>> Acesso em: 20 set. 2020

WITTEN, Ian H.; FRANK, Eibe; HALL, Mark A. **Data mining: practical machine learning tools and techniques**. 3 ed. Morgan Kaufmann, 2011.

YIN, Robert K. **Estudo de Caso: Planejamento e métodos**. Porto Alegre: Bookman, 2015.

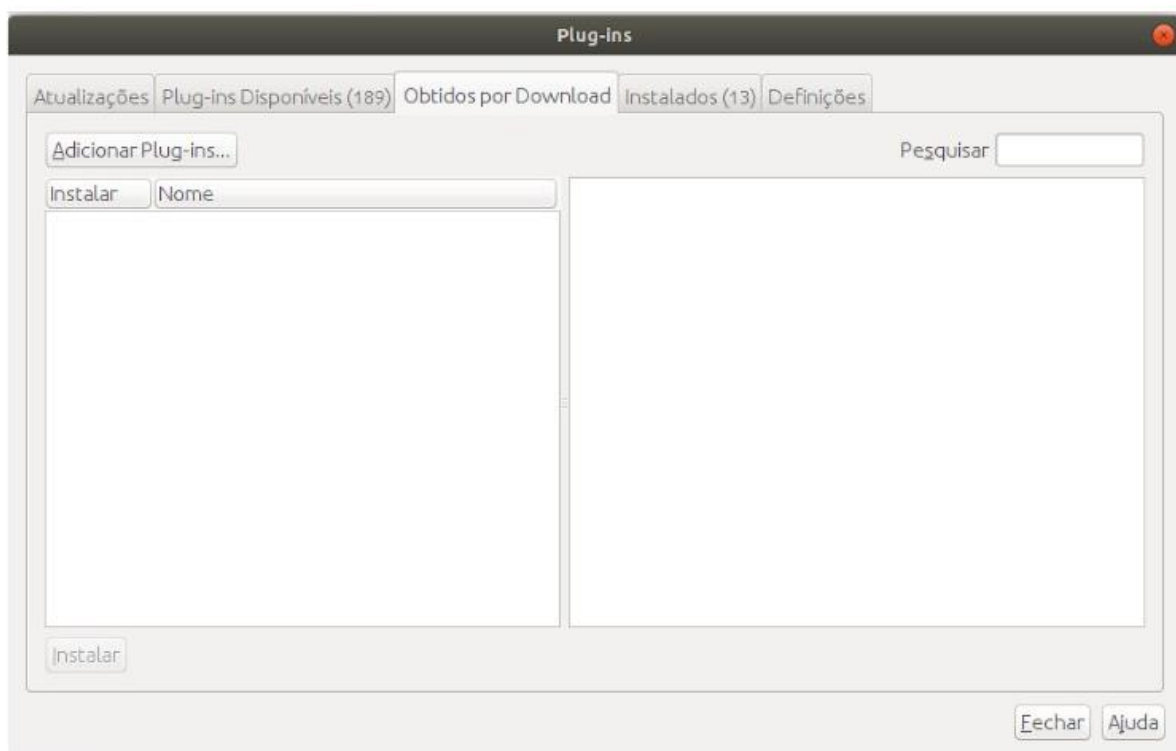


## APÊNDICE A – GUIA PARA INSTALAÇÃO E UTILIZAÇÃO DA FERRAMENTA “EASECHAT”

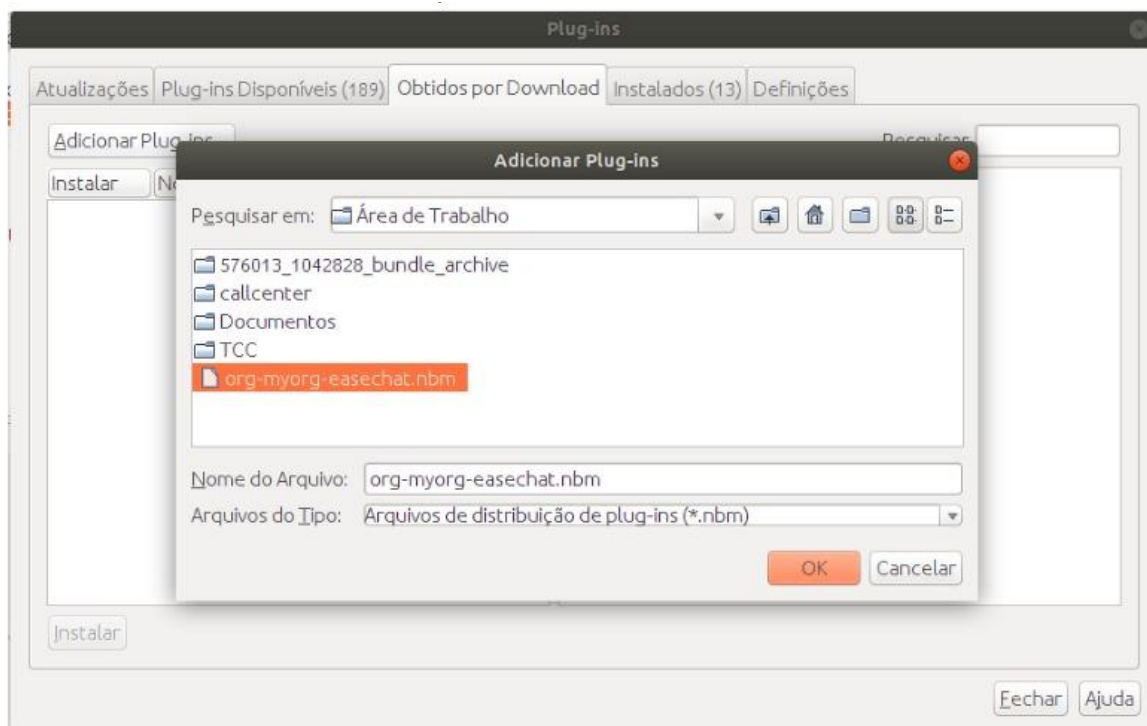
### Instalando e utilizando o plug-in “EaseChat”

Link para download no drive: <https://drive.google.com/file/d/1I44nEyBC55wcMAIk3RLYCj-QsttFmki/view>

Na barra superior do Netbeans IDE clique em: Ferramentas > Plug-ins  
Clique na opção “Adicionar plug-ins”

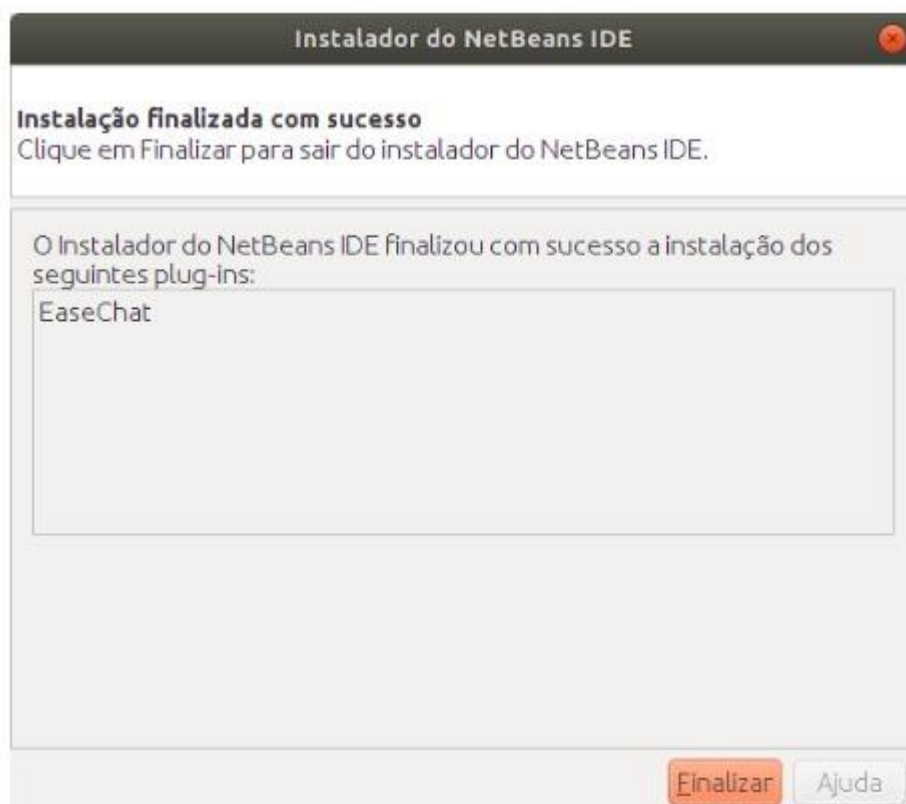
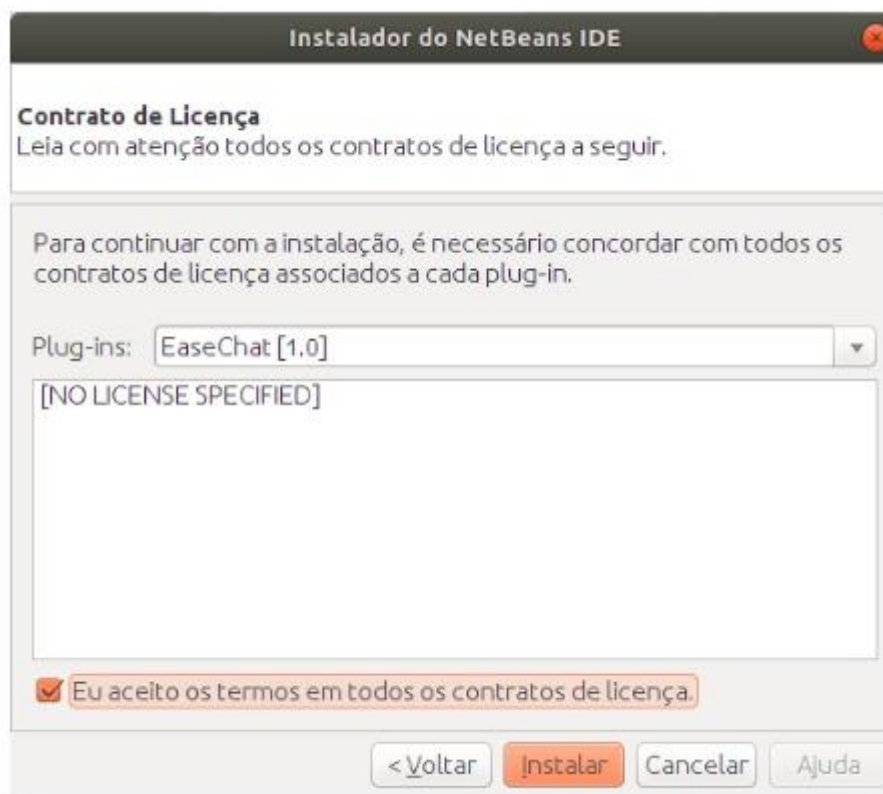


Em seguida abrirá a tela para você encontrar o local onde seu plug-in foi salvo. Quando encontrá-lo, selecione-o e clique em “ok”.

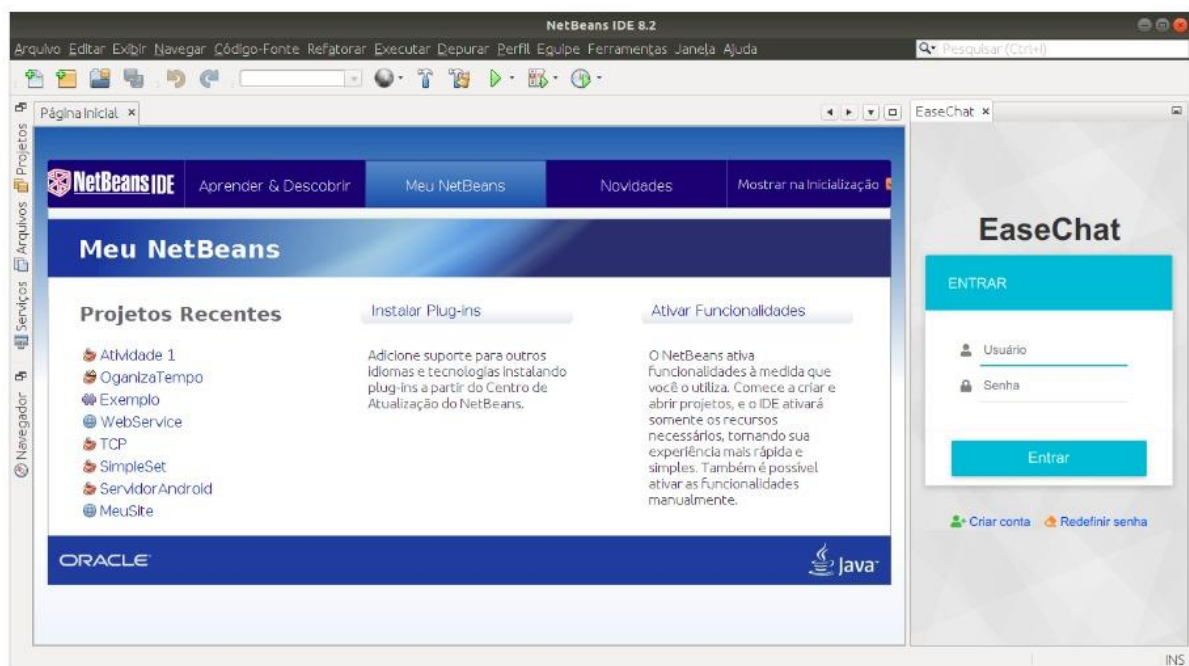


No instalador, basta seguir as etapas.





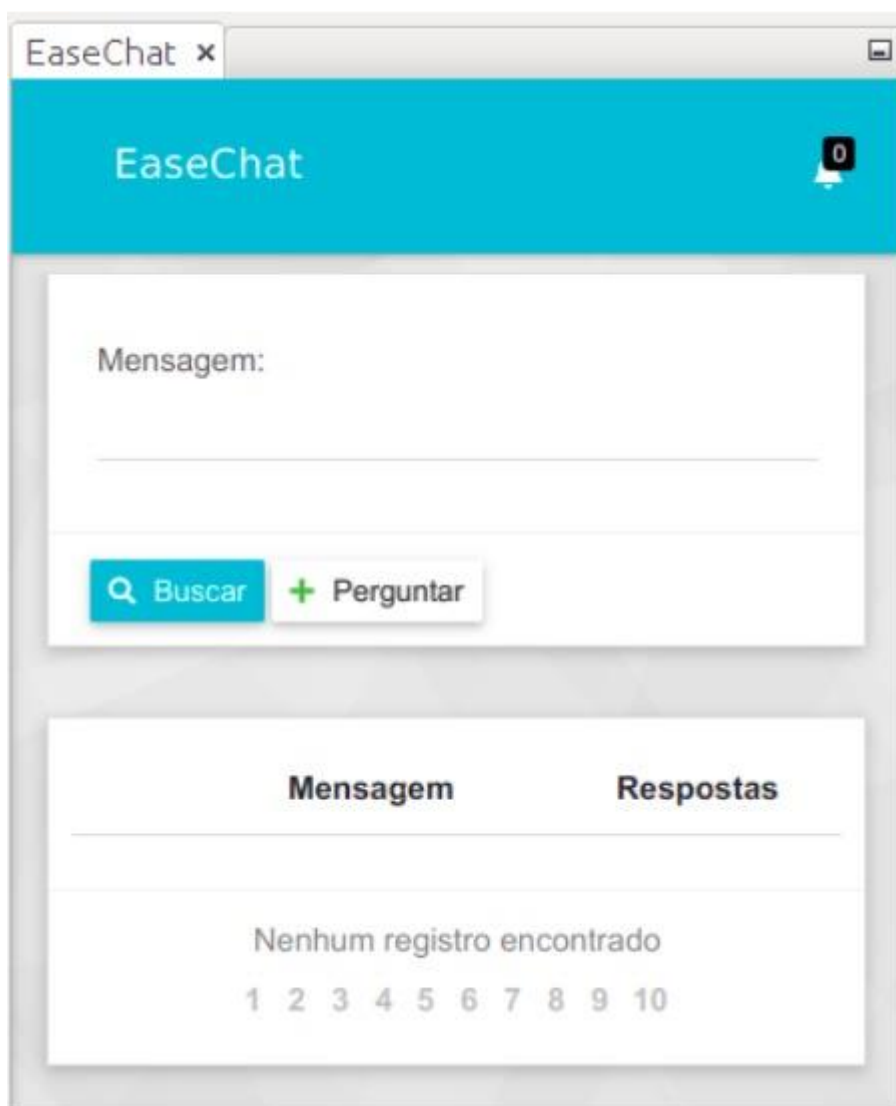
Após finalizar a instalação o plug-in já irá aparecer na IDE, mas você também pode acessá-lo no menu superior na aba Janela > EaseChat



Para realizar o login basta inserir seu usuário do email @universo.univates (Ex: **taina.schaeffer**@universo.univates.br) e a senha padrão: senha123

A senha padrão é temporária e poderá ser trocada na tela de login em “Redefinir senha”. Informe seu usuário e, em seguida, a nova senha.

Para enviar sua dúvida clique em “Perguntar”.



Nesta listagem de discussões você poderá ver as dúvidas dos seus colegas. Para ver mais informações, você pode clicar no ícone da lupa. Para fazer uma contribuição clique na setinha.

EaseChat

1

Mensagem:

Buscar

+ Perguntar

	Mensagem	Respostas
<div><div>Q</div><div>→</div></div>	Olá, estou com dúvidas na atividade 1	1

1 a 1 de 1 registros

1

2

3

4

5

6

7

8

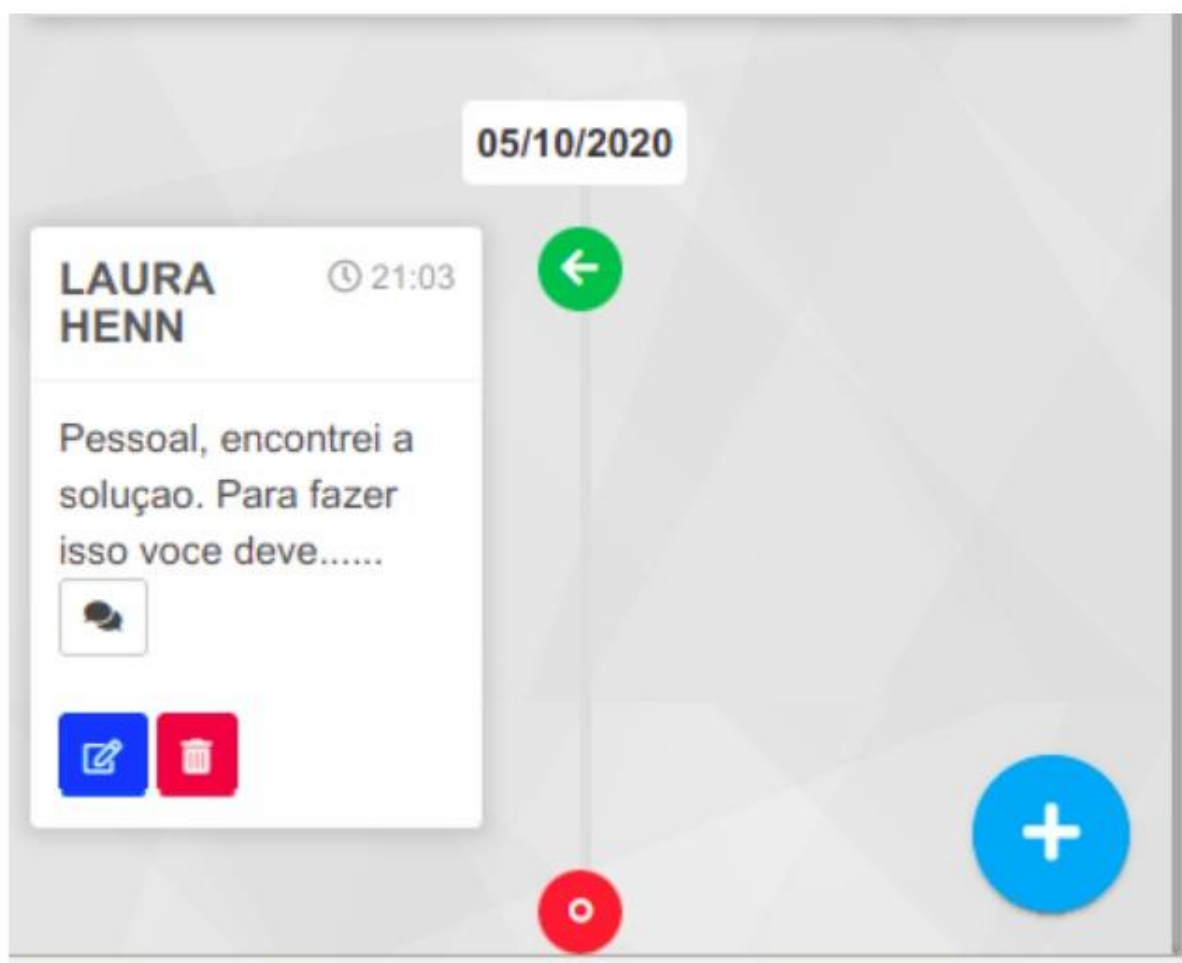
9

10

Para responder, clique no botão “+” e envie a sua contribuição.



Você pode enviar respostas para alguma dúvida que você compartilhou. Suas mensagens ficarão sempre na parte esquerda da tela e dos seus colegas na parte direita. Caso você queira editar a mensagem você pode clicar no botão azul e caso queira excluir seu comentário clique no botão vermelho.



**Ficou com alguma dúvida?**  
**[taina.schaeffer@universo.univates.br](mailto:taina.schaeffer@universo.univates.br)**



## **APÊNDICE B – QUESTIONÁRIO DE AVALIAÇÃO DO “EASECHAT”**

1 – Atribua uma nota para a importância de uma ferramenta colaborativa para apoio do aprendizado:

BAIXA – 1 ( ) 2 ( ) 3 ( ) 4 ( ) 5 ( ) – ALTA

2 - Atribua uma nota para a eficácia da proposta de uso da ferramenta:

BAIXA – 1 ( ) 2 ( ) 3 ( ) 4 ( ) 5 ( ) – ALTA

3 – Você acredita que o uso a longo prazo de uma ferramenta colaborativa pode auxiliar no seu aprendizado?

Sim ( )

Não ( )

4 – Você acredita que aprender em conjunto com outros colegas facilita o seu aprendizado?

Sim ( )

Não ( )

5 – Destaque pontos positivos da ferramenta

6 – Destaque pontos negativos da ferramenta